НАСТРОЙКА ПРОКСИ-СЕРВЕРОВ ДЛЯ UNIX И WINDOWS

Сеть для всех

Помегабайтная оплата Интернет при правильной организации доступа позволяет существенно сэкономить. Весь вопрос в выборе и правильной настройке программ

Денис Романюк, root@ampersand.cv.ua

В наше время недорогих выделенных линий, ISDN-модемов и радиоканалов WaveLan коллективный доступ в Интернет — вполне привычное дело. При этом оплата, в отличие от коммутируемых модемных соединений, осуществляется не повременно, а за объем трафика. Неудивительно, что при «выделенном» подключении рано или поздно встает проблема кэширования — ведь это реальный способ уменьшить внешний трафик и сэкономить на оплате соединения.

Файлы под рукой

Вкратце идею кэширования можно описать так. Когда с одного компьютера клиент выкачивает вэб-страницу или двоичный файл (фотографию, архив, исполняемый файл), то программное обеспечение интернет-шлюза не просто передает файл клиенту, но предварительно сохраняет его на своем жестком диске. И в случае, если другому клиенту с другого компьютера понадобится тот же файл, то ПО определяет, что этот файл уже сохранен, и не загружает его из Интернета, а выдает локальную копию с диска интернет-шлюза.

Функция кэширования интернет-трафика реализована, как правило, в прокси-серверах, которые называются «кэширующими». Напомним, прокси-сервер — это ПО, которое служит «посредником» для выхода в Интернет клиентов локальной сети. Оно отправляет в Сеть запросы «от своего имени» (с реальным IP-адресом интернет-шлюза), а принимая в ответ вэб-страницы и файлы, «раздает» их тем клиентским станциям, которые их запрашивали.

Практически все прокси-серверы, ориентированные на работу с большим числом локальных клиентов, имеют функцию кэширования. Из наиболее распространенных кэширующих прокси-серверов можно назвать коммерческий Microsoft Proxy Server, а также бесплатный Squid, работающий в Linux- и UNIX-системах и распространяемый по лицензии GPL.

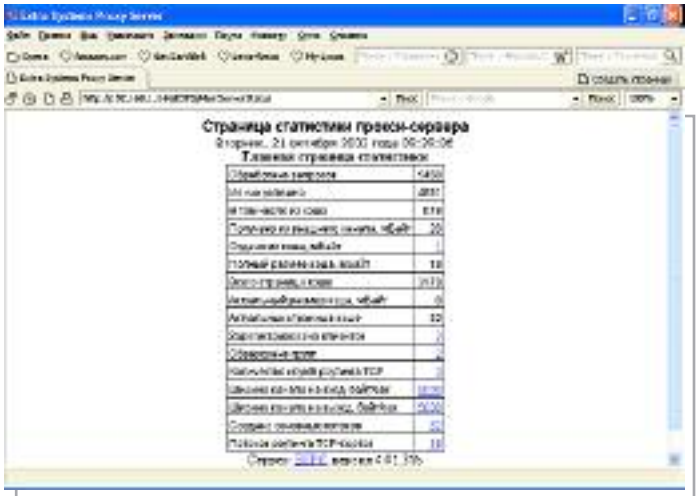
Использование кэширования чрезвычайно выгодно при помегабайтной оплате Интернет. В случае хорошего «попадания» в кеш, т.е. когда описанная выше ситуация с повторным запросом одного документа происходит довольно часто, внешний трафик может значительно сократиться.

Но даже в случае неограниченного подключения, когда оплата Интернета фиксирована и объем трафика не влияет на ее размер, прокси-сервер может быть полезен в смысле уменьшения загруженности внешнего канала. Кроме того, использование прокси-сервера может быть элегантным решением распространенной проблемы: «У меня есть интернет на одном компьютере в моей локальной сети, как мне разрешить выход туда всем прочим компьютерам?»

Пусть компьютер, имеющий прямое подключение к интернету (назовем его сервером или шлюзом), имеет адрес 192.168.1.144, а все другие компьютеры локальной сети — адреса типа 192.168.1.xxx. Наша задача заключается в том, чтобы установить и настроить на компьютере-шлюзе прокси-сервер, и соответствующим образом сконфигурировать клиентские компьютеры.

При выборе прокси-сервера

имеет смысл отдавать предпочтение программам с достаточной функциональностью и развитыми настройками — даже если в данный момент задача, стоящая перед вами, минимальна (выход клиентов в Вэб). Ведь очень скоро окажется, что вам, к примеру, совершенно необходимо обеспечить тонкое разграничение доступа по клиентам, или, скажем, запретить доступ к определенным серверам или URL и т. д.



Полная мобильность: с помощью Bluetooth отснятое фото передается на мобильный телефон

Ясно, что выбор программы-сервера зависит и от типа операционной системы, установленной на компьютере-шлюзе. Как правило, в качестве операционных систем для интернет-шлюзов используются Microsoft Windows 2000 Server, Microsoft Windows 2000/XP Workstation, а также бесплатные Linux и FreeBSD. Поэтому мы рассмотрим базовую настройку прокси-серверов для Windows 2000/XP и Linux/UNIX.

Непростой выбор

Существует большой выбор программ для Windows, имеющих прокси-функции: Microsoft Proxy Server, WinGate, WinProxy, Proxy+, DeleGate, FreeProxy и др. Однако большая часть этих программ не бесплатна, а ведь для домашней сети или малой фирмы дополнительные расходы нежелательны. Другие программы относятся к свободно распространяемому ПО, однако русскоязычная документация к ним либо скудна, либо отсутствует.

В качестве примера удачного бесплатного кэширующего прокси-сервера можно назвать отечественную программу ESPS (Extra Systems Proxy Server). На сайте разработчиков *www.esproxy.org.ua* можно найти документацию по конфигурированию (правда, некоторые вопросы администрирования ESPS лучше освещены на странице, посвященной предыдущей версии программы — *www.esps.org.ua*). Существует также группа новостей *news://news.extrasystems-freeware.org.ua/software.esps*, в которой обсуждается данный продукт.

В прокси-сервере ESPS, несмотря на его бесплатность, реализованы обширные возможности настройки, при том что его установка и конфигурирование не вызывают проблем. Он нормально пропускает вэб-трафик (протоколы HTTP и HTTPS) и умеет осуществлять маршрутизацию портов. Недостатком его является невозможность организации FTP-доступа с браузера — можно использовать лишь FTP-клиенты и файл-менеджеры (FAR, Total Commander и др.).

Вопросы установки

Текущая версия программы — ESPS 4.01 за июль 2003 года. ESPS может работать под любой версией Windows — включая 9x и NT, однако авторы не рекомендуют использовать ее под Windows 9x. Наилучшая платформа — Windows NT, 2000 или XP. Дальнейшее изложение предполагает, что вы используете именно эту платформу.

Кроме того, для повышения быстродействия создатели программы советуют не использовать сервер как обычный клиентский компьютер. Рекомендуется также размещать каталоги кэша на дисках NTFS, а не FAT32 и не забывать регулярно выполнять дефрагментацию диска.

Инсталляция программы нестандартна, но проста — загруженный с сервера файл нужно запустить из командной строки с параметром INSTALL (*Espsrv.exe /INSTALL*). При этом программа копируется в системный каталог *WINDOWS* (например, *C:\WINDOWS*) и регистрируется в системе как сервис с именем *ES Proxy*. Вы должны увидеть его, если заглянете в *Control panel|Administrative tools|Services*.

Обратите внимание на то, что тотчас после инсталляции сервис регистрируется, но НЕ запускается, поэтому вам нужно самостоятельно открыть окно *Services* (*Службы*) в Панели управления, щелкнуть правой кнопкой на ES Proxy и выбрать *Start*. Если вы не желаете этого делать, можете попросту перезагрузить компьютер, ибо ESPS по умолчанию запускается при каждой загрузке системы.

Кроме того, следует запомнить, что после любого изменения в конфигурации необходимо перезапустить сервис для того, чтобы изменения вступили в силу — для этого придется опять-таки открыть окно *Services*, найти службу ES Proxy, и выбрать команду *Restart*.

Особенности настройки

В момент первого запуска программа создает в системном каталоге Windows (*C:\WINDOWS*) файл *esps40.ini,* который и хранит все настройки сервера. Специфика программы ESPS заключается в том, что ее конфигурирование происходит в стиле UNIX — через редактирование текстовых файлов (тогда как большинство Windows-программ настраиваются посредством диалоговых окон). Это одна из причин того, что программа при очень приличной функциональности имеет столь малый размер (всего 100 КБ).

По ходу работы программа создаст в том же каталоге файлы с именами типа *esps4_*.dat* — обратите внимание, что расширение указывает на то, что файлы двоичные, т.е. не предназначены для того, чтобы в них что-либо изменять вручную. Поэтому не обращайте внимания на эти файлы.

Теперь приступим к конфигурированию — откроем в текстовом редакторе вышеупомянутый файл *esps40.ini* в каталоге WINDOWS. Как и любой INI-файл, он делится на секции, названия которых проставле-

ны в квадратных скобках. Секции состоят из пар «параметр-значение», разделенных знаком равенства.

Начнем с создания простейшей рабочей конфигурации ESPS. Для этого отредактируем и добавим некоторые параметры в секциях [*Connect*], [*Cache*] и [*Routers*]. Все прочие параметры оставим как есть.

```
...
[Connect]
Enable=1 ; разрешить https-соединения
...
[Cache]
Enable=1 ; включить кеширование
Path=F:\PROXY_CACHE      ; путь к
каталогам кеша
...
[Routers]
Enable=1 ; разрешить перенаправление
портов
List File
Name=D:\WINDOWS\esps4_routers.txt  ;
путь к файлу настройки портов
```

Обратите внимание, что после того, как вы установите *Enable=1* в секции [*Cache*] и первый раз перезапустите сервер, будут созданы каталоги кеша (компьютер в это время может немного притормаживать). По умолчанию каталоги кеша будут созданы в каталоге *WINDOWS\ESPS4 Cache*, поэтому перед перезапуском сервера проверьте наличие свободного пространства на системном разделе, так как создается 65536 вложенных каталогов, которые, даже будучи пустыми, занимают полгигабайта. Если же вы хотите, чтобы каталоги файлового кеша были созданы в другом месте, вам следует явно указать путь к ним в [*Cache*]-секции INI-файла, напр., *Path=F:\ESPS_CACHE* (см. листинг).

Персональный доступ

Кроме файла *esps40.ini*, следует отредактировать файл настройки пользователей. Путь к нему можно найти в секции [*Users*] файла *esps40.ini* (см. параметр *Groups File*). Создадим файл с таким именем, откроем его в редакторе и пропишем настройки для групп пользователей (листинг *esps4_users_list.txt* можно найти в полном тексте статьи на CHIP-CD).

В этом файле создано всего 2 группы — *Administrators* и *Users* (при желании вы можете создать и большее количество групп). Принадлежность компьютера к группе определяется его IP-адресом через команду *USER*. Причем если второй параметр команды *USER* — маска подсети — задана как *255.255.255.255*, то это означает, что данной командой к группе приписывается ОДИН компьютер с указанным адресом.

Если же маска определяется по-другому (напр., *255.255.255.0*), то к группе приписывается целая группа компьютеров, находящаяся в указанной сети. Например, команда «*USER : 192.168.1.0 : 255.255.255.0 :*» приписывает к группе все компьютеры с адресами *192.168.1.xxx* (не забудьте заменить адреса в примере на реальные адреса вашей сети).

Проход воспрещен!

Командой *Disable URL* мы можем запретить обращение ко всем адресам, соответствующим определенной маске. В нашем примере мы запретили все URL, содержащие в себе слова «porn» и «sex», а также содержащие слово «banner» и заканчивающиеся на «.gif». Например, если на странице, просматриваемой пользователем, будет картинка с адресом *http://reklamka22.ru/cgi/banner/qw.gif*, то она не будет загружена.

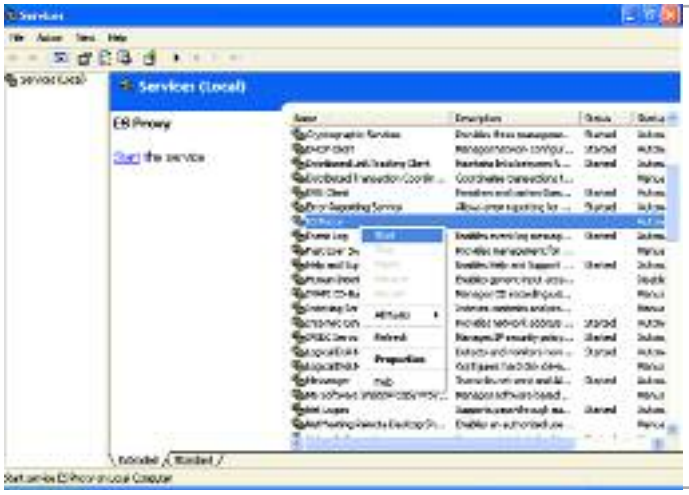
Теперь, когда мы внесли и сохранили изменения в файлах *esps40.ini* и *esps4_users_list.txt*, перезапустим сервис ESPS (через Control Panel — Services). После этого можно настраивать браузеры на клиентских компьютерах для выхода в интернет через созданный нами прокси-сервер (используется порт 3128, см. вставку на с. 100).

Не только вэб

При необходимости с помощью ESPS можно настроить маршрутизацию портов. Эта задача может возникнуть, например, тогда, когда клиентам в нашей локальной сети понадобится снимать почту с внешних POP3-серверов. Например, некоторые пользователи имеют почтовые аккаунты на сервере *pop3.rambler.ru*, другие — на сервере *ns.ampersand.cv.ua*. Пользователи хотят получать входящую почту с этих серверов прямо в своих почтовых клиентах (Outlook, The Bat и др.). Чтобы осуществить эту задачу, создадим файл *esps4_routers.txt* в корневом каталоге Windows и напишем в нем следующее:

```
;
ROUTER: POP3-Почта сервера на дру-
жественном сервере :
Remote Server : ns.ampersand.cv.ua :    ;
имя сервера
Remote Port   : 110 :                   ;
порт доступа на удаленном сервере
Local Port    : 110 :                   ; порт, от-
крываемый для этого сервиса на ло-
кальной машине
;_____
ROUTER     : Почта POP3 на Рамблере
:
Remote Server : pop3.rambler.ru :
Remote Port   : 110 :
Local Port    : 127 :
```

Как видим, в обоих случаях значение параметра *Remote Port* (удаленный порт) в обоих случаях равно 110, ибо именно этот порт по умолчанию отвечает за POP3-соединения. Что касается локального порта (*Local Port*), то он обязательно для каждого раздела ROUTER дол-



Полная мобильность: с помощью Bluetooth отснятое фото передается на мобильный телефон

жен быть разным, ибо этой командой мы открываем порт на нашем сервере (т.е., на компьютере, где выполняется ESPS). Поэтому в первом случае мы написали 110, а во втором — 127. Это совершенно произвольные числа — но нужно, чтобы эти порты не были уже заняты.

Далее необходимо внести изменения в *esps4_users_list.txt*, разрешив пользователям (через *Enable Route Port*) пользоваться 110-м и 127-м портами (см. листинг esps4__users__list.txt в тексте статьи на CHIP-CD). Сохранив изменения в обоих файлах, перезапустим сервис ESPS и будем настраивать почтовых клиентов.

Самое главное в настройке клиентов — не забыть установить номер порта POP3 в то значение, которое необходимо для маршрутизации на нужный сервер. Т.е., в нашем примере в случае маршрутизации на Рамблер в почтовом клиенте необходимо изменить номер порта POP3 на 127, а адресом POP3-сервера прописать машину с ESPS (192.168.1.144). Разумеется, вышеописанная операция возможна не только с серверами POP3 и 110-м портом. Например, подобным образом указываются внешние SMTP-сервера по 25-ому порту (адрес SMTP-сервера, как и адрес POP3-сервера, указывается в клиентской почтовой программе).

Учет и контроль

Группы пользователей, которым в конфигурационном файле *esps4_users_list.txt* разрешен просмотр статистики (значение *Show Statistics* равно 1), могут просмотреть ее, набрав *http://192.168.1.144/ESPS/MainServerStatus* в окне браузера, подключающегося через данный прокси сервер. Разумеется, *192.168.1.144* в указанном примере следует заменить на адрес прокси-сервера в вашей сети.

Страница статистики представляет собой таблицу, в которой некоторые числовые значения являются ссылками на другие страницы статистики. Наиболее интересна ссылка, стоящая в строке «*Зарегистрировано клиентов*» — она ведет на страницу учета траффика — т.е. сколько мегабайт выкачано каждым из клиентов за определенный период.

Дополнительные возможности

Разумеется, все нюансы работы ESPS выходят за рамки этой статьи. Из документации, выложенной на указанных выше сайтах, вы узнаете о нюансах тонкой настройки сервера: например, как установить лимит трафика для клиентов; как заставить ESPS вести протокол (лог) всех операций; как настроить различные правила кэширования для определенных URL (скажем, не закачивать ZIP-файлы объемом больше 100 МБ); как ограничить некоторым клиентам скорость доступа в Интернет и т.д.

Классика

В операционных системах Linux/UNIX бесспорным лидером является прокси-сервер squid, который эволюционирует в течение многих лет. Он достаточно компактен для не слишком мощных систем, обслуживающих небольшое число пользователей, и в то же время способен работать в корпоративных сетях с сотнями пользователей.

Перед тем, как скачивать и инсталлировать squid, убедимся, не установлен ли он уже у нас (для этого можно воспользоваться командами *which squid* либо *find / -name squid -type f -print* (первая команда работает быстрее, но может иногда выдать отрицательный результат несмотря на то, что сервер в системе присутствует). Если эти команды дали ненулевой результат, то значит, squid у вас уже установлен. В этом случае, памятуя лозунг врачей «не навреди», лучше опустить фазу установки и перейти непосредственно к конфигурированию.

Если же сервера squid в системе не найдено, скачаем последнюю стабильную версию 2.5 (*ftp://squid.nlanr.net/pub/squid-2/STABLE/squid-2.5.STABLE4.tar.gz*) и сохраним файл *squid-2.5.STABLE4.tar.gz* в каталоге /tmp. Затем перейдем в этот каталог и распакуем архив, после чего перейдем в появившийся каталог *squid-2.5.STABLE4* и выполним компиляцию и установку (это может занять некоторое время).

```
gzip -dc  squid-2.5.STABLE4.tar.gz | tar xvf -
cd squid-2.5.STABLE4
./configure; make all ; make install.
```

После выполнения этих команд squid будет проинсталлирован в */usr/local/squid*, однако надобно еще создать каталог, в котором squid будет хранить свой кеш и назначить владельцем каталога *squid* пользователя *nobody* (т.к. в целях безопасности squid по умолчанию выполняется с правами *nobody*):

```
mkdir /usr/local/squid/var/cache
chown -R nobody /usr/local/squid
```

Самый главный файл

Конфигурирование squid заключается в редактировании файла squid.conf, который будет находится в директории */usr/local/squid/etc/squid.conf* (если же squid был проинсталлирован из RPM-пакета, то он может оказаться в каталоге */etc/squid*). Следует помнить, что после каждого изменения в этом файле необходимо заставить squid перечитать свои конфигурационные файлы, что можно сделать, в зависимости от вашей версии Linux/UNIX и формы инсталляции squid, либо послав ему сигнал HUP (*killall -HUP squid*) либо просто через меню *service|squid|reload*.

Сразу же после инсталляции *squid.conf* будет содержать в себе значения по умолчанию (на случай, если вы захотите вернуться к этому варианту файла, в том же каталоге лежит *squid.conf.default*. В этом же файле можно познакомится с огромным количеством возможных директив конфигурации).

Однако файл *squid.conf*, устанавливающийся по умолчанию, имеет очень большой размер из-за множества закоментированных строк (т.е., строк, начинающихся с символа #). Чтобы получить тот же файл без комментариев, можно выполнить команду grep . squid.conf.default | grep -v '^#' > squid.conf, после чего файл *squid.conf* будет состоять всего из нескольких строк, в которых уже не так трудно разобраться.

Начнем с простого

Рекомендуется для начала использовать простейший вариант squid.conf (см. листинг squid.conf в тесте статьи на CHIP-CD). В дальнейшем при желании вы можете подстраивать squid под свои нужды.

Приведенная на CHIP-CD конфигурация разрешает для локальной сети

(*192.168.1.xxx*) и для адреса *127.0.0.1* подключения по протоколам HTTP, HTTPS и FTP, но запрещает доступ к URL, содержащим слова «porn», «sex», а также содержащие слово «banner» и заканчивающиеся на «.gif».

Списки контроля доступа

Сначала в файле перечисляются так называемые списки контроля доступа (*acl*, Access Control Lists), в которых могут быть указаны как IP-адреса клиентов, так и порты доступа или запрашиваемые URL. В последнем случае используются регулярные выражения (например, banner.*\.gif$), с помощью которых можно формировать сложные шаблоны запрашиваемых документов. Регулярные выражения — отдельная и непростая тема, однако даже начинающему системному администратору Linux/UNIX желательно изучить основы их применения, поскольку это чрезвычайно мощный и полезный инструмент.

После списков доступа (*acl*) перечисляются правила доступа (*http_access*) с директивой резрешения/запрещения (*allow/deny*) и со ссылкой на соответствующий список доступа.

Закончив первичную настройку файла *squid.conf*, необходимо запустить сервер squid. Однако перед первым запуском необходимо создать директории кеша, запустив исполнимый файл squid с аргументом «-z»: */usr/local/squid/sbin/squid -z*. Теперь можно запускать программу: */usr/local/squid/sbin/squid -D* либо просто *service squid start* (и не забудем прописать это в одном из стартовых сценариев, чтобы squid запускался при каждой перезагрузке системы). После этого squid будет слушать свой порт 3128 и отвечать на запросы.

Разумеется, для того, чтобы все это работало, правила нашего брандмауэра (см. ЧИП 10/2001, с. 118) должны разрешать входящие подключения на порт 3128 для адресов нашей локальной сети, а также исходящие запросы в Интернет на 80, 443, 20, 21, 53-й порты с той машины, на которой происталлирован squid.

Учет и контроль

По умолчанию squid ведет протокол всех операций, log-файлы при этом находятся в каталоге */usr/local/squid/var/logs* (либо */var/log/squid*). Анализируя простеньким скриптом файл *access.log* из этого каталога можно организовать учет трафика по клиентам (см. ЧИП 2/2003, с. 98). Однако необходимо следить за тем, чтобы лог-файлы не переполняли диск.

Место для роста

Обратите внимание, что squid — это сложная в настройке, но очень мощная программа, способная выполнить почти всякую мыслимую задачу в пределах компетенции прокси-сервера. Например, его можно настроить так, что он будет пускать пользователей в интернет только после ввода пароля; можно научить squid пользоваться кэшем других прокси-серверов (например, сервера вашего провайдера); он может ограничивать скорость доступа в интернет определенным пользователям.

Сервер squid может применяться также для кеширования исходящего веб-трафика, тем самым ускоряя и разгружая вэб-сервер. Возможности интернет-шлюза с сервером squid парктически неограничены — нужно лишь правильно настроить его и обеспечить адекватным «железом».

Настройка клиентов

После конфигурирования и запуска proxy-сервера необходимо настроить браузеры на клиентских компьютерах с адресами *192.168.1.xxx*. Адрес прокси-сервера в примере *192.168.1.144*.

Internet Explorer
В меню *Tools|Internet options|Connections|LAN settings* включим опцию *Use a proxy server*. Введем в соответствующих полях IP-адрес (*192.168.1.144*) и порт (*3128*) нашего прокси-сервера (см. рис. внизу). При необходимости тонкой настройки нужно щелкнуть на кнопке *Advanced*.

ESPS, squid и многие другие прокси-серверы обычно используют по умолчанию порт 3128

Для Opera
меню *File|Preferences|Network*, вкладка *Proxy servers*, включить опции HTTP, HTTPS и FTP, написав адрес и указав порт прокси-сервера (см. рис. справа) Настройка прочих браузеров осуществляется аналогично.

Squid поддерживает FTP-доступ из браузера — нужно лишь указать в браузере порт прокси

Разумеется, не только браузеры могут выходить в Интернет через прокси. Например, если прокси-сервер является единственными воротами в Интернет, то даже ICQ на клиентских компьютерах можно заставить выходить в онлайн через прокси.
Для этого прокси-сервер должен пропускать https, а в ICQ надо открыть *Main|Preferences|Connections|Servers* и выбрать *Using firewall* и *Using proxy*.
В выпадающем списке выбрать протокол HTTPS. В строке *host* должно быть *login.icq.com*, а в строке *port* — 443.
После этого щелкнуть на закладке *firewall* окна настроек *Preferences*, выбрать в списке HTTPS и ввести IP-адрес и порт нашего прокси-сервера.

INFO

Еще о прокси-серверах

ЧИП 2/2003, с. 98, ЧИП 10/2001, с. 118
CHIP 1/2004 Текст статьи с листингами конфигурационных файлов
http://links.chip.ua/cache