small http сервер — обзор возможностей

крис касперски, ака мыщъх, no-email

small http – это компактный (~100 кб в архиве), бесплатный, шустрый http/htts/ftp/pop3/smtp/dns/dhcp/proxy сервер в одном флаконе, поддерживающий SSL/TLS и портированный практически на все платформы: NT, BSD, Linux. это настоящий швейцарский нож, удобный в администрировании и идеально подходящий для SOHO-рынка, но, к сожалению, недостаточно известный и совершенно затерянный на фоне кучи конкурирующих с ним пакетов, большинство из которых проигрывают ему по всем статьям. впрочем, конечное решение о выборе сервера, остается за читателем…

введение
/ полоса 1, колонка 1 */*

Большинство мелких компаний используют довольно тяжеловесную артиллерию в лице MS IIS/Apache, 99% возможностей которых оказываются никак не востребованными и которые предъявляют к квалификации администратора явно завышенные требования, не говоря уже о системных ресурсах. Железо просто ложится под нагрузкой, а пользователи получают информацию в час по чайной ложке. Ситуация усугубляется тем, что MS IIS/Apache — очень сложные программные продукты, состоящие из миллионов строк кода, в которых то и дело обнаруживаются ошибки, приводящие к возможности удаленных атак на сервер, для предотвращения которых администратор вынужден постоянно быть в курсе последних событий и неустанно скачивать свежие заплатки. Скажите честно — оно вам надо?!

Small HTTP хорош тем, что, будучи локальным проектом местного масштаба, он выпадает из поля зрения хакеров, концентрирующихся вокруг крупных программных пакетов. Конечно, никто не говорит, что Small HTTP свободен от ошибок, но, во всяком случае, в открытом доступе нет ни одного exploit'a и навряд ли они появятся в дальнейшем. Так что пользователи Small HTTP могут чувствовать себя как у Христа за пазухой.

К тому же, Small HTTP обладает весьма продвинутым функционалом, которым может похвастаться далеко не каждый сервер. Он поддерживает IPv6, удаленное управление, ведет детальные логи, автоматически обнаруживает большинство типовых атак (т.е. берет на себя функции IDS), блокируя нарушителей или предоставляя это сделать администратору, в результате чего необходимость в установке брандмауэра/системы обнаружения вторжений сразу же отпадает.

Small HTTP покрывает по меньшей мере 99% типовых потребностей SOHO-рынка и потому установка дополнительных программных пакетов _не_ требуется. К тому же, в отличии от того же IIS, Small HTTP реализован на всех популярных платформах и если в один "прекрасный" день вы решите перейти с Windows на BSD/Linux, то… вам потребуется просто скопировать текстовой конфигурационный файл на новую машину и все!!! Пользователи, пароли, виртуальные директории, черные списки IP-адресов, политика proxy-сервера — восстановится сама собой! А вот переход с IIS на Apache требует немыслимых затрат времени и, зачастую, кардинальной перестройки HTML-кода.

Вы уже заинтригованы?! Тогда вперед!

развертывание Small HTTP сервера
/ полоса 1, колонка 2 */*

Идем на http://smallsrv.com/ (см. рис. 1) и в разделе "Download" выбираем заархивированную версию сервера под нашу платформу, например, под Windows. Скачиваем архив, представляющий в данном случае исполняемый файл, который мы и запускам на выполнение. Появится диалоговое окно инсталлятора (см. рис. 2), в котором нам предлагают выбрать запуск сервера в качестве обычного процесса (по умолчанию) или же системного сервиса, для чего нужно взвести галочку напротив пункта "NT Service" (требует прав администратора). Разницы, в общем-то никакой. Работать будет и так и эдак.

Чтобы сервер автоматически загружался вместе с системой, взводим галочку "Add to startup". Наконец, соглашаемся с лицензионным соглашением, назначаем пароль администратора и нажимаем на "Install". После чего, в системном трее появляется симпатичная

иконка, щелкнув по которой левой клавишей мыши мы открываем окно монитора, позволяющее нам наблюдать чем в данный момент занят наш подопечный (см. рис. 3).

Теперь поговорим о регистрации. Официальная цена сервера — 49$, что меньше чем совсем ничего, то есть практически даром. Для граждан СНГ стоимость лицензии снижена до 300 рублей (при условии коммерческого использования). Просто относим платежную квитанцию в банк и наслаждаемся жизнью. Если же на кассе пусто, можно зарегистрироваться и бесплатно. Для этого в правом вернем углу официальной странички выбираем "rus", заходим в раздел документации (русскоязычной, разумеется) находим имя пользователя "Гражданин бывшего СНГ" и код активации, меняющийся от версии к версии. Кстати говоря, при попытке зарегистрировать свою версию сервера данным образом, автор столкнулся с упорным отказом сервера "переваривать" указанный в документации код активации и разработчику Small HTTP было отправлено письмо с вопросом "что я делаю не так?". Оказалось, что автор все делал правильно, просто в его версию сервера вкралась ошибка, приводящая к невозможности бесплатной регистрации, которую разработчик тут же исправил, возвратив письмо со ссылкой на обновленную версию, зарегистрированную без проблем и танцев с бубном.

Однако, бесплатная регистрация — крайняя мера и автор, очарованный возможностями Small HTTP сервера, в конечном итоге все-таки решился заплатить за него обозначенную сумму, хоть и использовал сервер в абсолютно некоммерческих целях (для распространения собственных статей и как http-proxy, обеспечивающий выход из домашней локальной сети в Интернет).

/* полоса 1, колонка 3 */

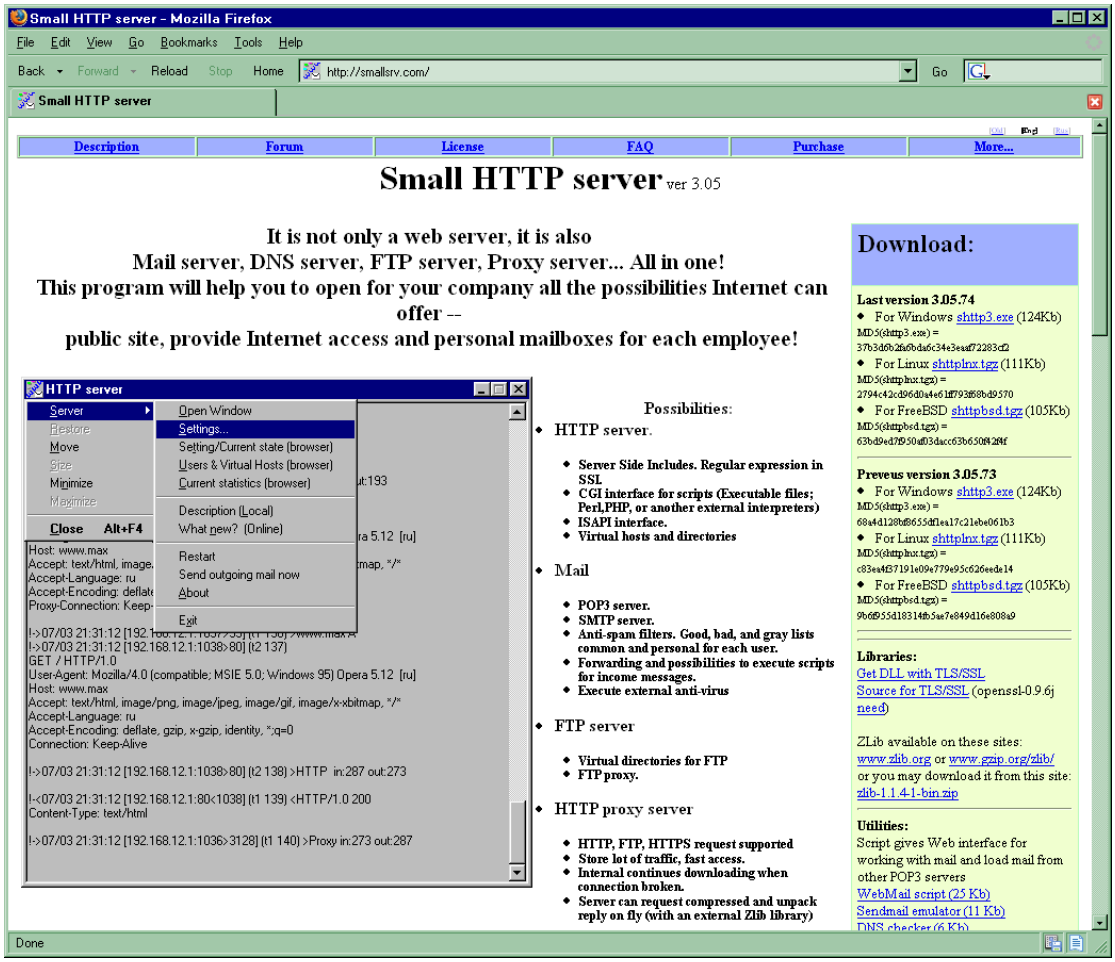

Рисунок 1 официальная страница Small HTTP сервера

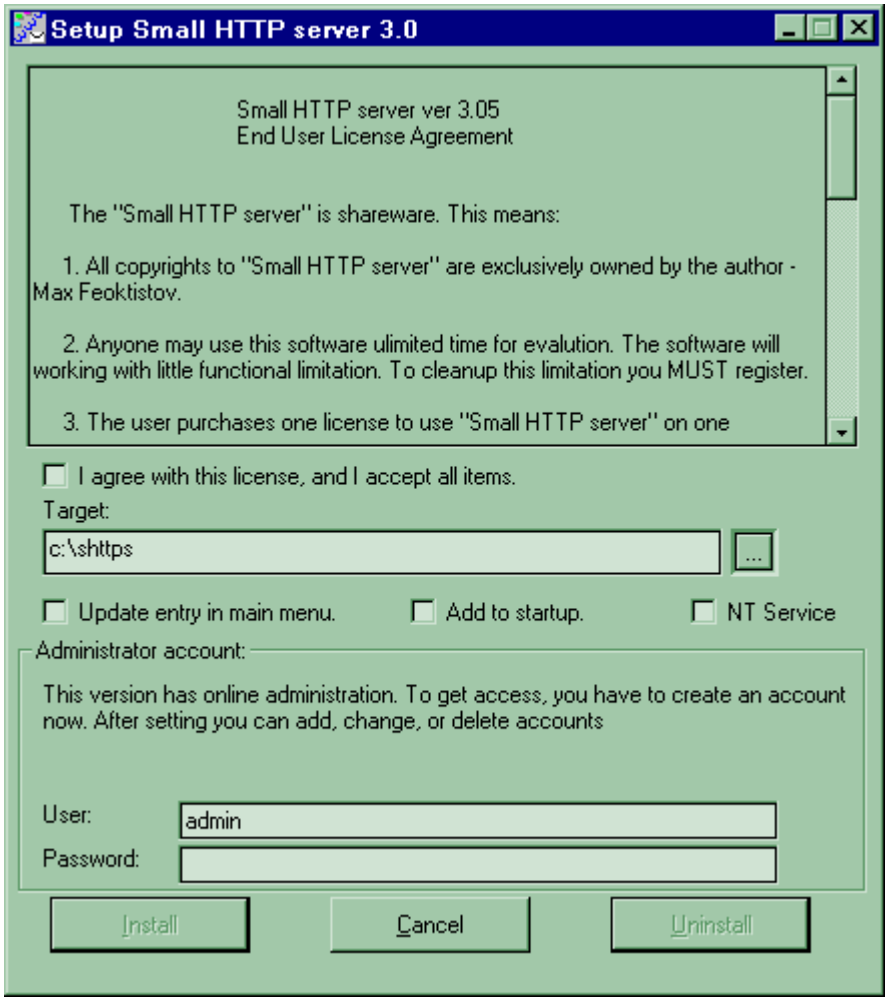

Рисунок 2 установка Small HTTP сервера под Windows

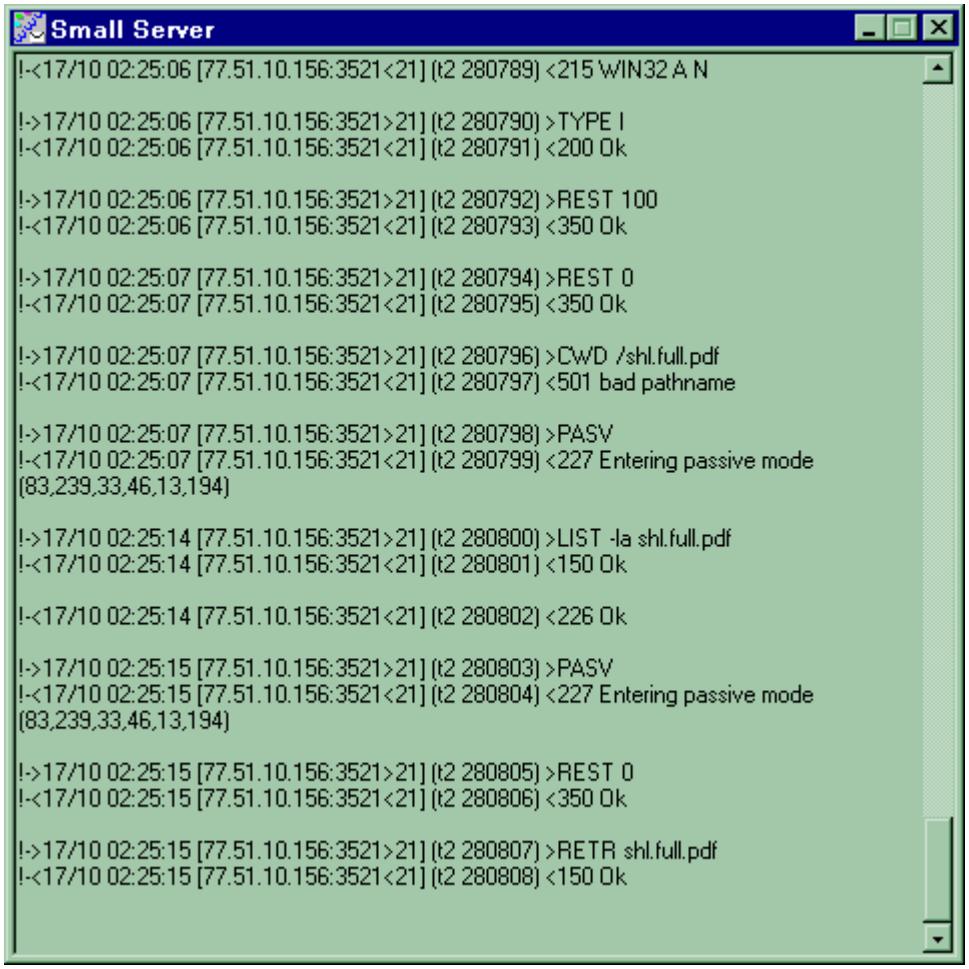

Рисунок 3 мониторинг текущей активности сервера

советы по настройке Small HTTP
/ полоса 2, колонка 1, 2 */*

Прежде, чем сервер начнет работать должным образом, его необходимо настроить. Щелкаем по иконке в трее правой клавишей мыши и во всплывшем контекстом меню выбираем пункт "Setting", после чего в появившемся диалоговом окне нажимаем на раздел "General" ("общие настройки" — см. рис. 4).

Взводим галочку "Created detailed log for…", заставляя сервер создавать детальный лог, позволяющий не только собирать статистику посещаемости, но и выявлять вандалов, стремящихся надругаться над нашим сервером. Только учтите, что детальные логи пожирают дисковое пространство со скоростью Хумахера, так что о нет следует позаботится заранее. Так же можно взвести галочку "Create a daily log", чтобы каждый день создавался новый лог — это существенно упрощает подчистку логов и разбор полетов.

Для достижения наивысшей производительности рекомендуется снять все ограничения по скорости отдачи файлов и количеству подключений (галочка "no restrict the number of simultaneous from each host"), однако, это следует делать только в том случае, если у вас толстый канал, а посещаемость сравнительно невелика, в противном же случае, любой придурок, запустивший многопоточную качалку, посадит всех остальных пользователей на голодный паек. С другой стороны, многие браузеры ведут себя очень странным образом, если сервер отказывается устанавливать требуемое количество соединений, что приводит к ошибкам отображения web-страниц, а это намного хуже, чем низкая скорость отдачи! Так что из двух зол приходится выбирать что-то одно.

Прокрутив окно настроек на несколько строчек вниз, мы обнаружим пункты, ответственные за "белые" и "черные" списки для IP4/IP6 адресов, действующие на все типы сервисов: proxy, http, ftp, pop3, smtp. Как нетрудно догадаться, эта функция позволяет отказаться от брандмауэра, создавая необходимые политики непосредственно на самом сервере. Но здесь

есть одна тонкость. Если брандмауэр наглухо блокирует заданный IP, то Small HTTP сервер развернуто объясняет ему, почему он не прав и должен ходить отсюда. А это — трафик. Добраться к закрытой информации злоумышленник, конечно же, не доберется, но забить канал мусором – это без проблем. Так что на случай целенаправленных атак, лучше все-таки держать брандмауэр под рукой (впрочем, достаточно многие провайдеры предоставляют услугу блокирования IP-адресов на своем маршрутизаторе, освобождая нас от необходимости оплачивать "хакерский" трафик).

Галочка "Enable web-administration", как и следует из ее названия, задействует удаленное управление Small HTTP сервером по HTTP протоколу через браузер. Естественно, для этого необходимо знать пароль администратора, назначенный при установке сервера, так что хакеры отдыхают, но все-таки лучше перестраховаться, ограничив список IP-адресов с которых разрешено подключение, перечислив их в строках редактирования, расположенных ниже. Только не забудьте включить туда подсеть своего провайдера и сотового оператора, чтобы иметь возможность управлять сервером из дома или непосредственно с помощью мобильного телефона, а не тащиться из-за каждой мелочи в офис.

Теперь переходим к настройкам HTTP-сервера, которые чрезвычайно обширны и разнообразны, а потому, если вы не чувствуете себя достаточно подготовленным, лучше ничего не трогать, воспользовавшись значениями по умолчанию. Единственное, что стоит изменить — это максимальное количество одновременных подключений с одного IP (чтобы не пользовали многопоточную закачку) и откорректировать предельную скорость отдачи, рассчитанную в соответствии с толщиной вашего канала деленной на среднее количество посетителей в произвольно взятый момент времени. За это отвечают пункты: "Limits for summary speed of outgoing transfer for all connections from the same IP" и "How many another connections must have activity, to check on speed limitation".

В конфигурации по умолчанию, HTTP-сервер отдает несжатый трафик, что не есть хорошо, особенно если HTML-страницы не перегружены графикой и состоят преимущественно из хорошо сжимаемого текста. Так почему бы его и не сжать?! Возвращаемся к официальной странице, скачиваем библиотеку zlib (http://smallsrv.com/zlib-1.1.4-1-bin.zip), копируем ее в любой каталог (например, в каталог, в котором установлен Small HTTP сервер) и указываем к ней путь в поле "DLL library Zlib", не забывая взвести галочку "Use gzip packing, if possible", после чего скорость отдачи сразу же возрастет в несколько раз, и пользователи, сидящие на медленных каналах, конкретно возрадуются, да и ваш канал разгружается. Так что глупо не пользоваться такой возможностью. Правда, сжатие требует дополнительных тактов ЦП и на слабых машинах с толстыми каналами выигрыш рискует превратиться в проигрыш и тут все решает эксперимент. Плюс, в библиотеке ZLib периодически обнаруживаются ошибки, ведущие к возможности захвата управления сервером, так что… тут есть над чем подумать.

/* полоса 2, колонка 3 */

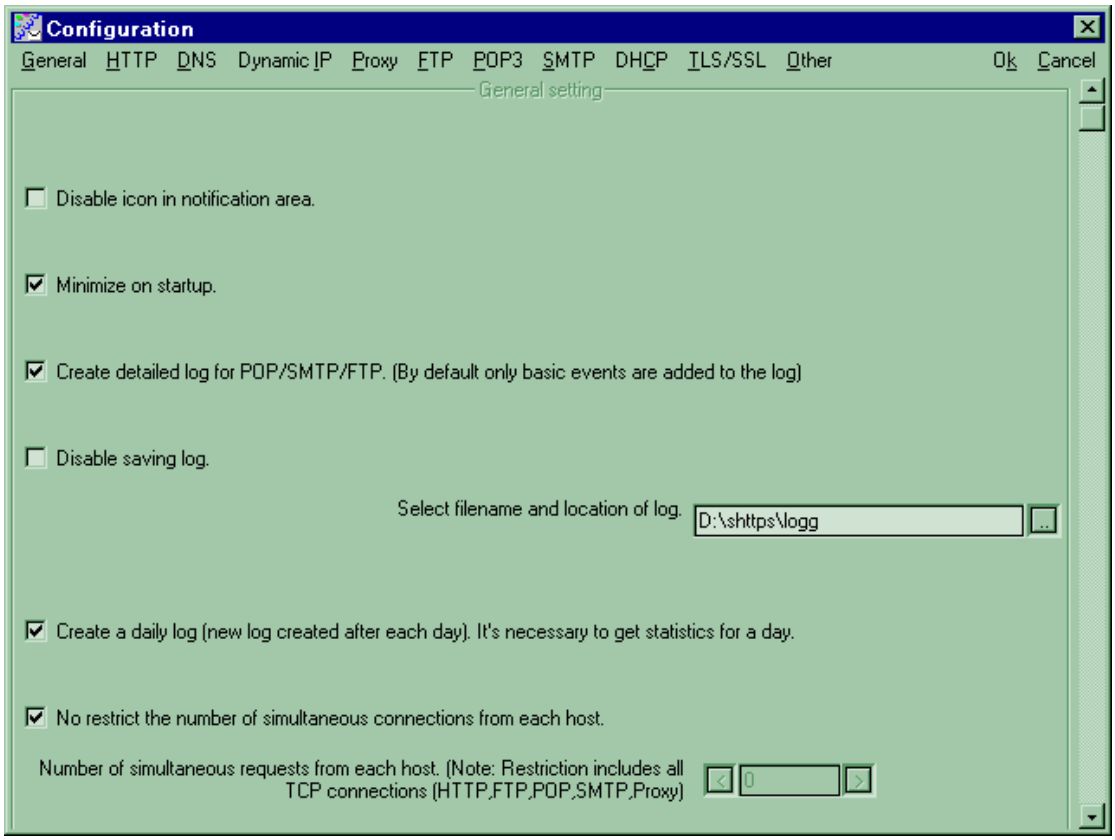

Рисунок 4 основные настройки Small HTTP сервера

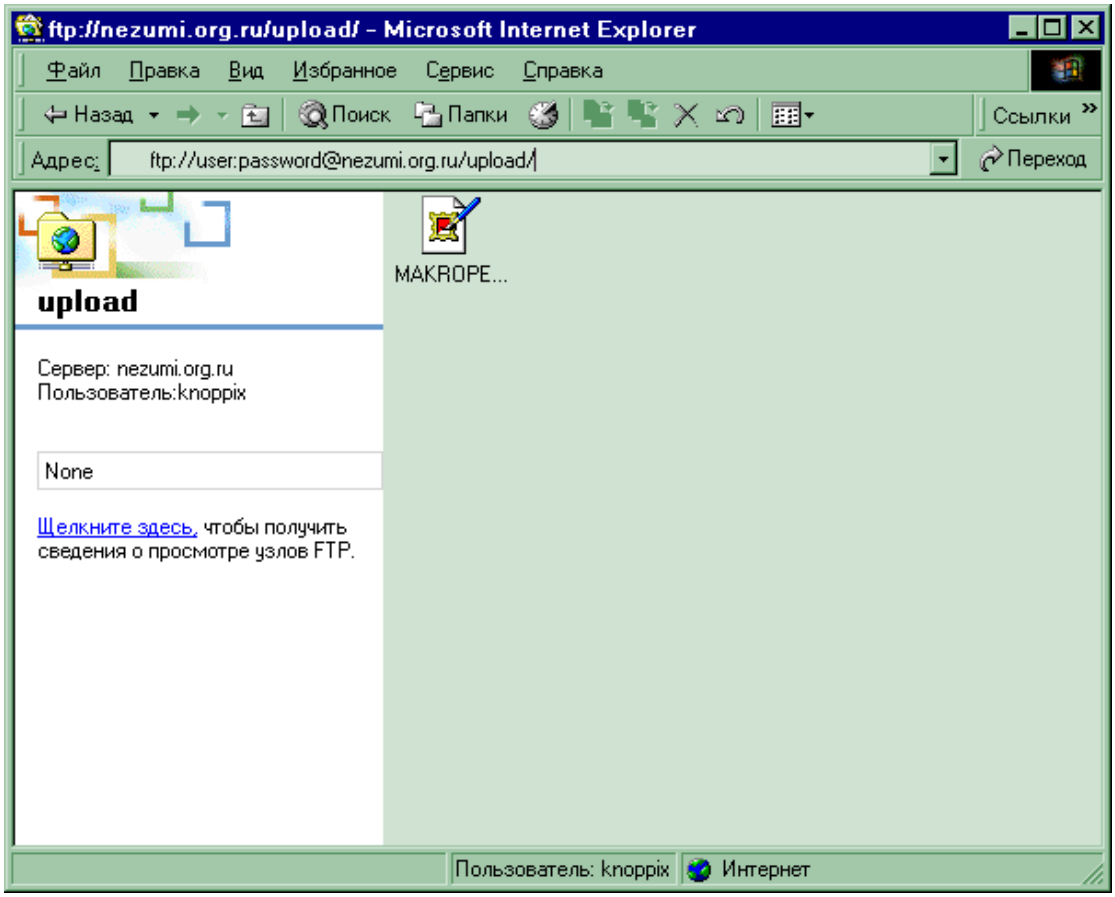

Рисунок 5 IE, используемый в качестве ftp-клиента

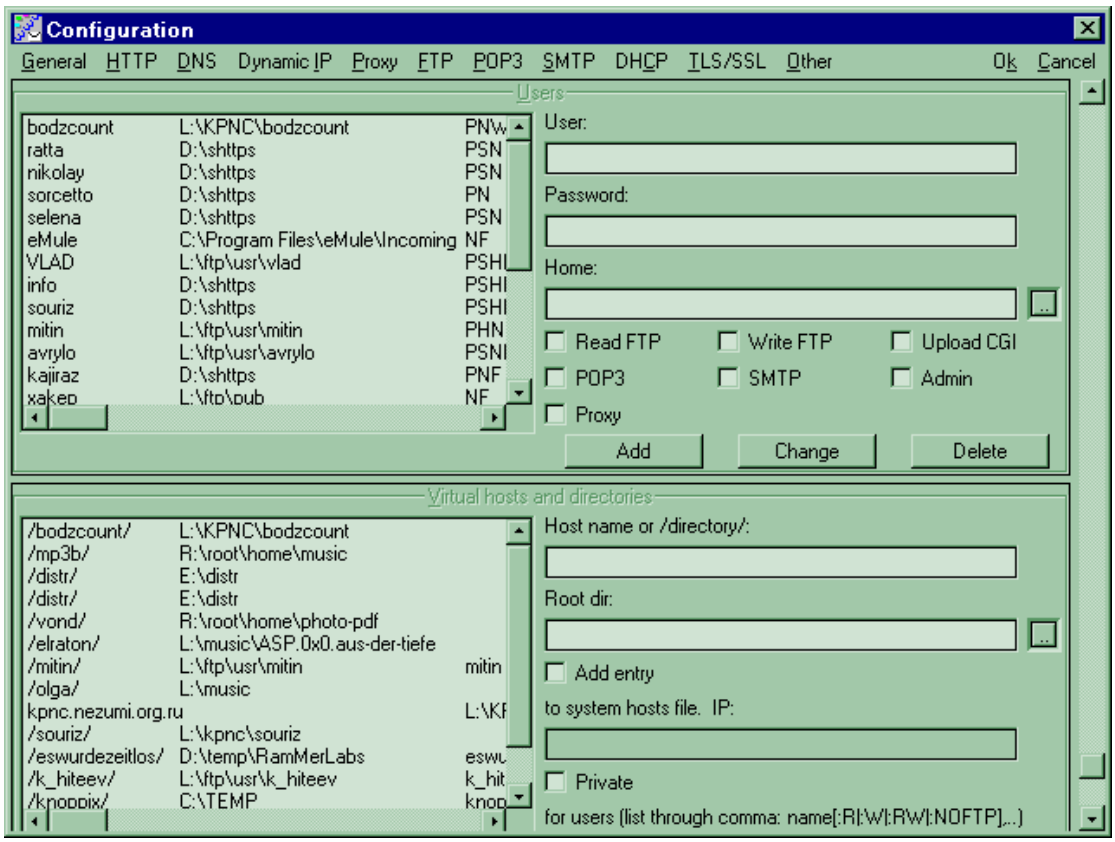

Рисунок 6 пользователи и пароли

воздвигаем собственный FTP
/ полоса 3, колонка 1 */*

Устойчивое заблуждение гласит, что среднестатистический пользователь совершенно не умеет пользоваться FTP-серверами, поскольку для этого требуется устанавливать специальную программу — FTP-клиента. И, чтобы ни у кого не возникало проблем, обмен файлами лучше всего осуществлять через электронную почту. Ну… можно, конечно и через почту, но это тоже самое, что забивать гвозди фотоаппаратом.

Вот вполне типичная ситуация: ваш партнер хочет передать вам файл мегабайт под пятьдесят, а то и побольше. Вопрос: и какой же почтовый ящик выдержит такой размер? Особенно если учесть, что при перекодировке вложений их размер может увеличиваться в 1,3 или 1,6 раз — это уже от почтового клиента отправителя зависит. Для справки, SMTP-сервер, встроенный в Small HTTP, ограничивает максимальный размер вложений 12 Мбайтами, а большинство бесплатных серверов не дают и этого, вынуждая отправителя разрезать файл на множество частей, что требует умения обращаться с RAR'ом (или другим архиватором подобного типа). А получатель потом матерясь собирает все эти фрагменты воедино, обнаруживая что часть из них дублируются, а часть — отсутствует.

На самом деле, FTP-клиент встроен практически во все браузеры, в том числе и в IE и, что самое главное, IE поддерживает закачку на FTP-сервер, выполняя ее совершенно прозрачно для пользователя. Просто заводим на сервере учетную запись с именем и паролем, формируем следующий URL – ftp://user:password@ftp.my-company.com/upload/ и посылаем его отправителю в электронном письме. Тогда ему останется всего лишь кликнуть по ссылке, дождаться открытия IE и перетащить туда файл (файлы) мышью — как они тут же окажутся на нашем FTP (см. рис. 5). Ну разве не крысота?! Одним махом решается целая куча проблем!

Естественно, FTP-сервер должен быть соответствующим образом настроен. Галочка "Enable FTP PORT" по умолчанию сброшена, что в ряде случаев приводит к невозможности установки соединения и закачке файла, поэтому, рекомендуется ее взвести.

Пользователи и пароли задаются в разделе Other → Users, где необходимо выбрать имя пользователя, назначить ему пароль, указать путь к папке, в которую будут падать заливаемые файлы и взвести галочки "Read FTP" и "Write FTP" (см. рис. 6). Однако, следует помнить, что открывая FTP-сервер на upload (даже с паролем!) мы серьезно рискуем, поскольку исходящий трафик по обыкновению бесплатен, а входящий — платен и если наш партнер разозлиться на нас, то ему ничего не стоит закачать своп-файл или другой мусор на много гигабайт, реально кидающий нас на бабки. К сожалению, Small HTTP не поддерживает квотирования, зато его поддерживает Windows 2000 и более старшие системы. Просто запускам Small HTTP от имени выделенного пользователя, огранивая его дисковое пространство с учетом _реальных_ потребностей. Тогда хакерская атака угаснет как бычок в писсуаре.

воздвигаем собственный DNS
/ полоса 3, колонка 2 */*

Практически все провайдеры предоставляют клиентам DNS-сервера, переводящие доменные имена в IP-адреса, и частенько эти DNS работают так, что лучше они бы вообще не работали… Тормоза, ошибки поиска заведомо существующих узлом, хакерские атаки, перенаправляющие нас на другой узел вместо запрошенного — вот неполный список проблем, с которым приходится постоянно сталкиваться администраторам.

Решение состоит в воздвижении своего собственного DNS-сервера (в роли которого может вполне выступать Small HTTP), напрямую обращающегося к корневым DNS-серверам. И хотя в некоторых книжках написано, что обращение к корневым серверам — процедура не быстрая и серьезно замедляющая WEB-сервфинг, по ходу дела все обстоит с точностью до наоборот: Small HTTP кэширует DNS-запросы, так что задержка возникает лишь при первом посещении данного ресурса, а все последующие запросы обрабатываются локальным сервером практически мгновенно, то есть _намного_ быстрее, чем DNS-сервером провайдера.

Короче, будем считать, что я вас заинтриговал. Лезем в настройки HTTP сервера, выбираем пункт "DNS", находим там поле "Size of cache for names (in bytes)" (Размер кэша имен в байтах) и увеличиваем его на сколько не жалко. Чем больше размер кэша, тем больше доменных имен в нем поместиться и тем реже будут происходить повторные обращения к корневым DNS-серверам. (Изменение размера кэша возымеет действие только после перезапуска Small HTTP сервера — просто закройте его, а потом запустите вновь).

Самое главное — необходимо взвести галочку "Enable DNS over TCP" (Использовать TCP протокол для DNS-запросов), что с 99,999% гарантирует невозможность создания подложного пакета, который бы DNS сервер воспринял как правильный.

В поле "Host file. File with host names and IP address for DNS server" (Хост файл. Файл с доменными именами и IP-адресами, используемый DNS-сервером) указываем путь к хост-фалу. Хост-файл имеет достаточно сложную структуру и если вы не считаете себя достаточно компетентным, чтобы составлять его самостоятельно, воспользуйтесь готовым рецептом, приведенном в листинге 1.

```
.                   IN  NS a.root-servers.net
a.root-servers.net IN  A  198.41.0.4
.                   IN  NS b.root-servers.net
b.root-servers.net IN  A  128.9.0.107
.                   IN  NS c.root-servers.net
c.root-servers.net IN  A  192.33.4.12
.                   IN  NS d.root-servers.net
d.root-servers.net IN  A  128.8.10.90
.                   IN  NS e.root-servers.net
e.root-servers.net IN  A  192.203.230.10
.                   IN  NS f.root-servers.net
f.root-servers.net IN  A  192.5.5.241
.                   IN  NS g.root-servers.net
g.root-servers.net IN  A  192.112.36.4
.                   IN  NS h.root-servers.net
h.root-servers.net IN  A  128.63.2.53
```

Листинг 1 строки, которые необходимо добавить в host-файл, чтобы наш DNS-сервер обращался за разрешением запросов к корневым DNS-серверам

заключение
/ полоса 3, колонка 3 */*

Возможности Small HTTP сервера на этом, разумеется не ограничиваются. Мы рассмотрели всего лишь несколько основных функций, а их там… намного больше, чем можно подумать. Так что чтение документации всячески приветствуется. Разобраться с сервером методом тыка (то есть через визуальный интерфейс) невозможно уже хотя бы потому, что далеко не все функции имеют визуальную репрезентацию. Одна командная строка чего стоит…

Конечно, Small HTTP сервер не лишен недостатков. В частности, он сильно "проседает" под большой нагрузкой и плохо справляется с наплывом посетителей. Он не поддерживает автоматическое балансирование нагрузки и кластеризацию (что умеют Apache и IIS), а потому проекты типа www.wikipedia.org на базе Small HTTP сервера воздвигнуть ни за что не получится. Только ведь никто их и не собирается воздвигать. Мы же говорим о SOHO-сфере, чьи сервера в основном работают как proxy и почтовики, а на web-странички заглядывают только конкуренты, да и то не из любопытства, а по долгу служебной необходимости.

Никакой другой программный продукт не обеспечит тех же возможностей на аналогичных условиях и в этом плане Small HTTP сервер просто уникален. И это не просто слова! Прежде чем их написать, автор перепробовал огромное количество программ, но всякий раз оставался недоволен — то сервер не показывает статистику, то не дружит с удаленным администрированием, то устанавливает драконические ограничения на максимальное количество подключений (в бесплатной версии), то не поддерживает других жизненно необходимых возможностей, вынуждая прибегать к помощи дополнительных программных пакетов, каждый из которых требует ухода (настройки, установки заплаток) и жрет системные ресурсы, вынуждая вкладывать деньги в железо только затем, чтобы прокормить всю эту ораву программ.

Естественно, о вкусах не спорят и навязывать свои предпочтения автор никому не собирается, но прежде чем отказываться от Small HTTP сервера необходимо знать от чего собственно вы отказываетесь. Почему бы не попробовать установить его в качестве secondary сервера, работающего на нестандартных портах (благо Small HTTP позволяет их менять), и не "проиграться" с ним некоторое время? Small HTTP – очень уживчивая штука. Одно время на машине автора был установлен десяток серверов и посетители nezumi.org.ru даже не подозревали с каким из серверов они работают в настоящий момент.

Теперь же из всех серверов остались только три: WarFTP, Etlin HTTP Proxy и Small HTTP. Это не реклама. Так… информация к размышлению.