Резервирование WEB сайта

Фленов Михаил aka Horrific http://www.vr-online.ru

Я думаю, что резервированием содержимого компьютера занимаются не более 10% ю-зверей ПК. Нет, у меня нет точных данных, я просто так думаю. Дело в том, что данные резервируют опытные пользователи, которые хоть раз их теряли благодаря выходу из строя железа или благодаря вирусам. Резервированием сайтов занимаются вообще единицы. Если не считать корпоративные серверы, а домашние страницы, то количество пользователей следящих за данными можно пересчитать по пальцам. Я сам раньше забивал на это дело, потому что не хочется тратить время и драгоценный трафик. Но это ошибка.

Опасность для сайта

Я уже более пяти лет развиваю свой сайт, но ни разу еще не резервировал его. Действительно, зачем это делать, если у моего хостера есть ежедневное резервирование, и он отвечает своими зубами и коронками за качество и доступность сервера. Если выйдет из строя железо, то его быстро восстановят с резервной копии, а если верить рекламе, то и восстанавливать ничего не надо, потому что на серваке используется RAID с зеркалированием. Вирусы хостеру также не страшны, потому что сервера работают под управлением FreeBSD, вирусы для которого можно пересчитать по пальцам. Вроде бы все отлично, потому что сайт находиться в надежной и безопасной зоне, как на военной базе. Но не все так прекрасно.

С ростом посещаемости сайта появляются не доброкачественные или просто бракованные посетители, которые так и стремятся нарушить целостность данных. Таких людей принято называть хакерами, но я бы назвал их по другому. От таких людей резервирование хостера не помогает. Если ты потеряешь базу данных, то потом задолбаешься просить хостера восстановить ее, поэтому лучше самостоятельно позаботиться о своих данных.

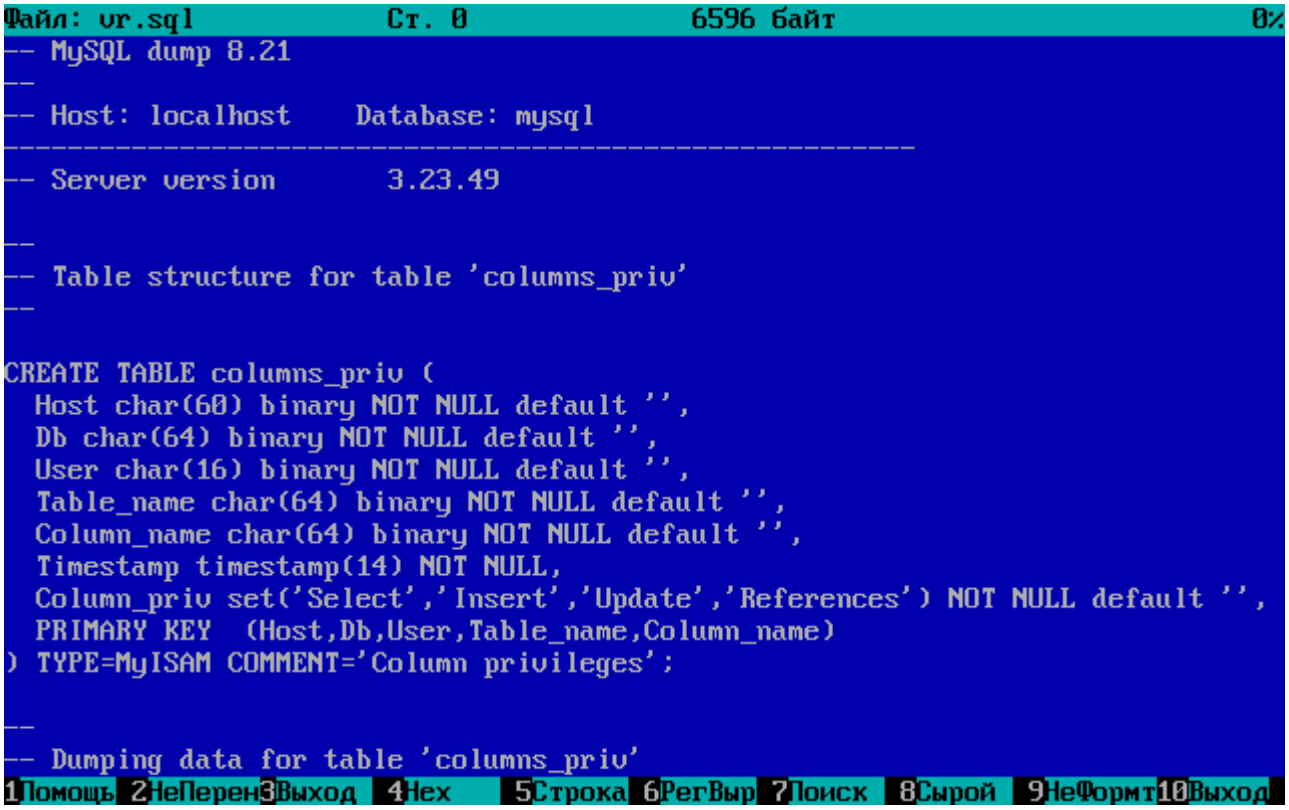
То же самое случилось и с моим ресурсом. Совсем недавно я поплатился за свою доверчивость к посетителям и потерял одну из баз MySQL. Чтобы восстановить ее понадобилось три дня, и все это время часть функций была не доступной. Да, именно базы данных чаще всего становятся проблемным звеном, поэтому мы начнем с этой темы.

Резерв из шелла

Если есть возможность и права на выполнение команд, то можно создать дамп базы с помощью утилиты mysqldump, которая входит в поставку сервера. Эта команда выполняется в следующем виде:

```
mysqldump параметры имя_базы > файл.sql
```

В результате, в файле файл.sql будет находиться дамп базы в виде SQL запросов. Чтобы восстановить структуру базы и данные достаточно выполнить запросы из файла и база данных окажется на родине.



Результат выполнения команды mysqldump

В качестве параметров необходимо как минимум указать ключ –p. В ответ на это, mysqldump запросит пароль пользователя, с которым нужно подключиться. По умолчанию используется имя пользователя root и если у этого пользователя пароль не пустой (а я надеюсь, что он не пустой, иначе даже дамп не поможет от хакеров), то без ключа –p утилита mysqldump не сможет подключиться к базе и сделать ее дамп.

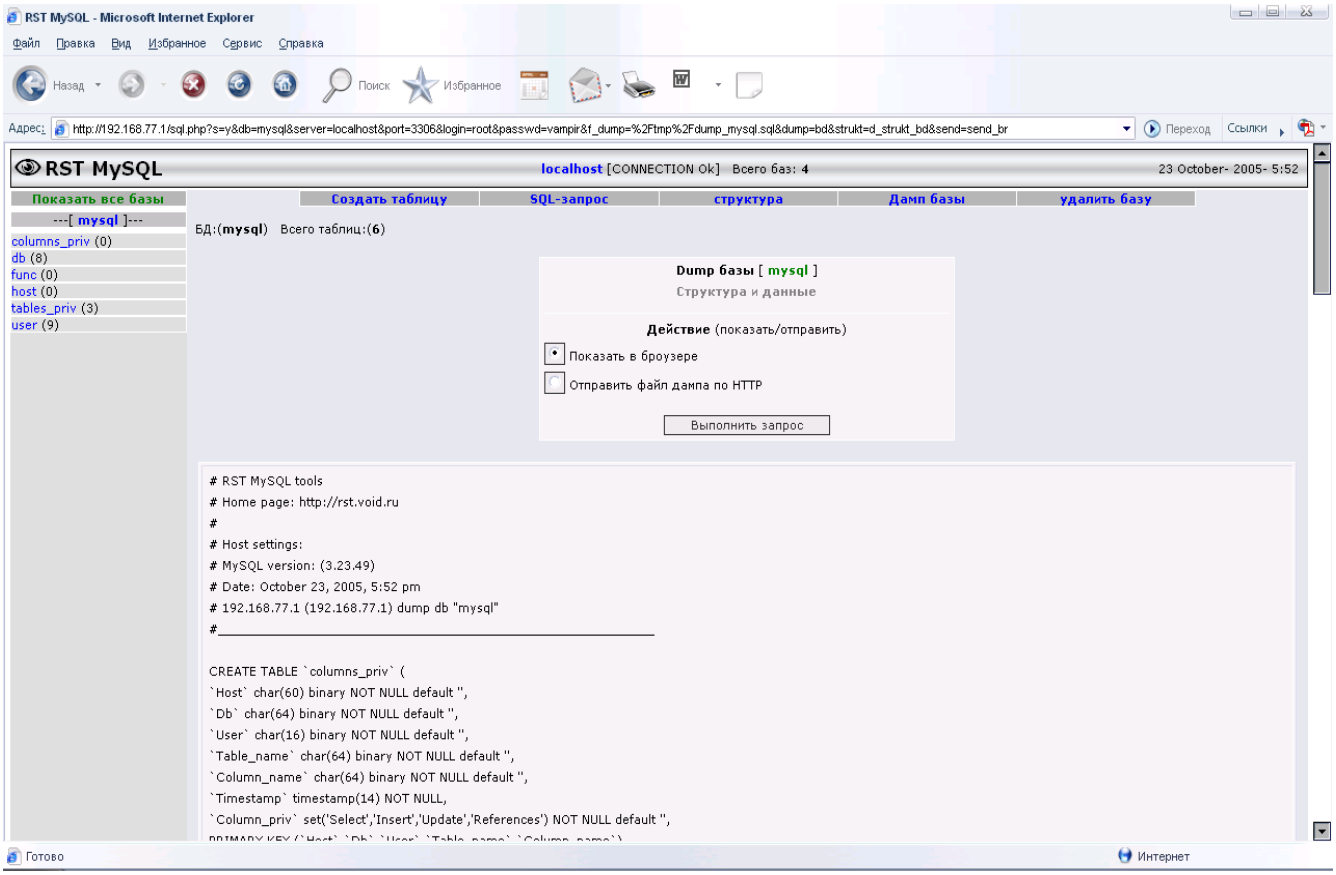
Итак, чтобы создать дамп базы данных mysite необходимо выполнить следующую команду:

```
mysqldump -p mysite > dump.sql
```

В ответ на это перед нами появиться запрос на ввод пароля. Если пароль указан верно, то в файле dump.sql появиться полный дамп базы данных mysite. Как было сказано выше, для восстановления теперь достаточно только выполнить все команды из этого файла и структура базы вместе с данными тут же вернуться на родину.

WEB дамп

Если нет возможности выполнять команду mysqldump (нет шелла или нет прав на выполнение утилиты, а может ее удалил злостный админ), то на помощь придут WEB утилиты управления MySQL сервером. Да, будем рассматривать именно этот сервер баз данных, как самый распространенный на площадках хостверов, и не только российских. Если у хостера уже есть установленный сценарий управления MySQL, то можно использовать его, а если нет, то рекомендую обратить свое внимание на RST MySQL или CyD MySQL Admin Centre.



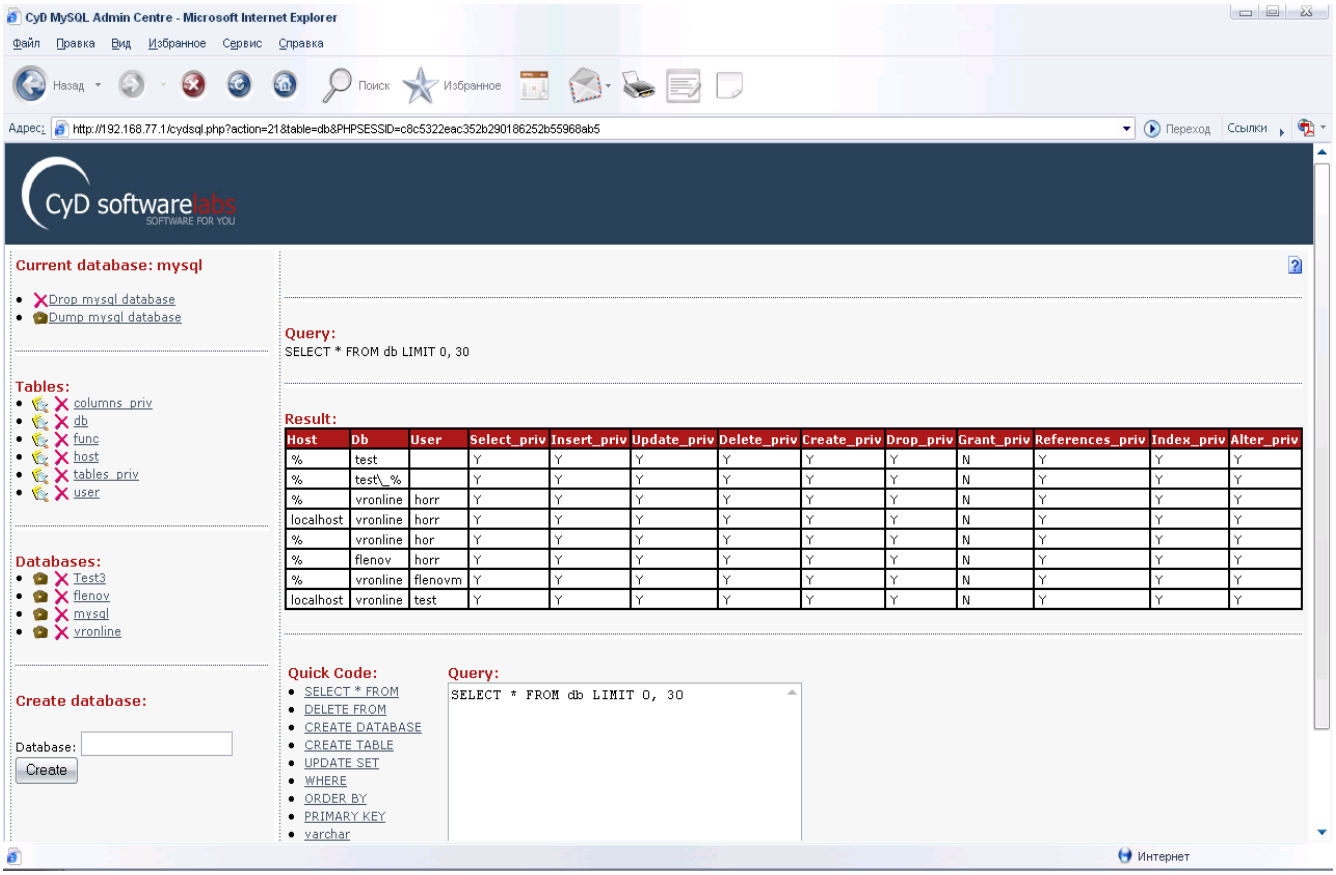
Управление MySQL сервером через сценарий RST MySQL

Первый (RST MySQL) является релизом всем известного сайта по безопасности void.ru, и взять сценарий можно с сайта http://rst.void.ru или с нашего компакт диска. Это действительно хороший и мощный сценарий, который состоит всего из одного файла и при этом, обладающий очень маленьким размером. Текущая версия занимает всего 81 кило и обладает всеми необходимыми функциями, как для администратора, так и для хакера. В том числе, есть возможность быстрого создания дампа базы.

Второй (CyD MySQL Admin Centre) является моим релизом, который пока в разработке и в Интернете еще не доступен, но на компакте можно увидеть и протестировать. Почему я взялся за разработку подобного сценария? Просто то, что я видел, является мощными сценариями, но не очень удобными и не очень красивыми. Я же не люблю использовать программу, если она не удобна и не красива, а так как я по природе (уж таким меня мама родила) программист, то начал писать свою реализацию. К тому же все сценарии халявные и я на этом не зарабатываю.

В CyD MySQL Admin Centre также реализована возможность создания дампа любой базы данных, лишь бы у тебя были права на выполнение необходимых сценариев, иначе можно получить зависоном по окну WEB браузера.

Вот тут надо дать пояснение. Может возникнуть подозрение, что я рекламирую свою работу. Нет, если бы это была реклама, то я не упомянул бы RST, а я уважаю void.ru и мне нравятся их релизы. Я только предлагаю альтернативу, а тебе выбирать, что удобнее.



Управление MySQL сервером с помощью сценария CyD MySQL Admin Centre

Если же ты не нашел ничего удобного, то можешь посмотреть листинг где-то рядом с этим текстом, где показан пример на языке PHP, который реализует дамп. Вот тебе листинг в руки, можешь написать собственный сценарий и сделать его таким, какой нужен тебе.

Потеря файлов

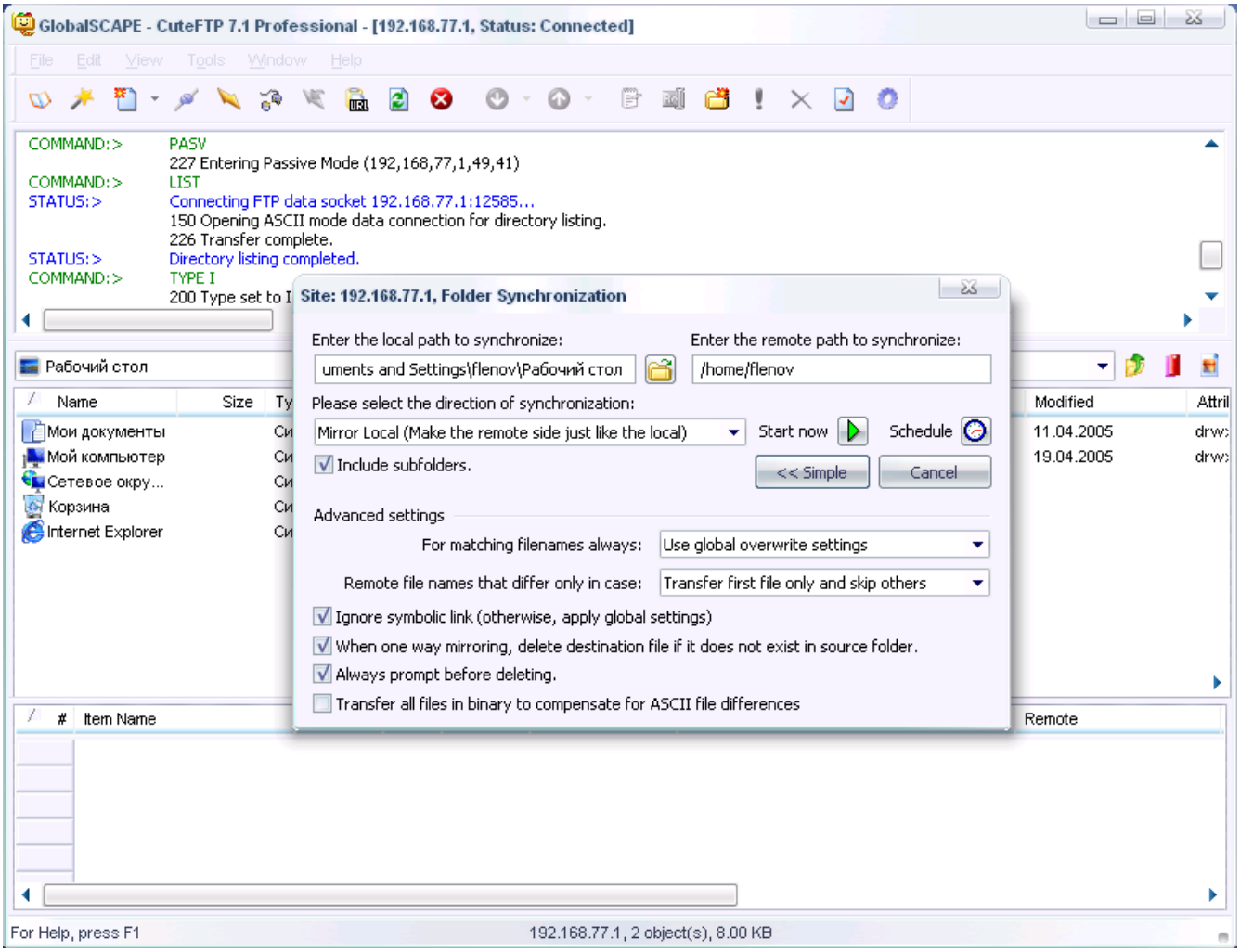
Не базой единой живет сайт. Есть еще файлы, которые также могут быть разрушены и хакеры нередко стремятся нарушить их целостность или просто удалить их. Сейчас я расскажу два случая, которые подтолкнули меня на поиск хорошей утилиты для восстановления файлов сайта.

Случай первый. Опять же, во время последнего взлома моего сайта на сервер был подкинут Shell, в качестве которого выступал релиз от void.ru с измененным заголовком. Я искал его несколько дней, потому что в файле журнала был такой xxxx, что найти в нем нужные записи было сложно, даже через фильтры, поэтому на разбор журнала ушло несколько дней. Конечно, потом я нашел нужные записи и быстро уничтожил чужой файл, но где гарантия, что нет другого шелла, который просто еще не запускался и не засветился в журнале? Проверять весь сайт, который состоит из тысяч файлов и сложную структуру директорий просто не реально, поэтому необходимо какое-то средство автоматизации поиска заразы, а антивирусы подобные вещи выполнить не могут.

Еще один случай – недавно мне пришлось восстанавливать сайт друзей после взлома и искать обидчиков и первым делом (до начала восстановления), я, конечно же, начал

сканировать сервер на наличие подкинутых файлов. Ничего лишнего на сервере не было, но через день взлом повторился. Я снова проверил все, но результат пустой. На сервере шелла не было. Точнее, не было отдельного файла, который работал бы как шелл. Хакер поступил интереснее – в один из сценариев был добавлен JavaScript код, который по щелчку в определенном месте отображал форму для ввода команд. Команды передавались функции system и не трудно догадаться, чем это грозит. Лишних файлов не было, поэтому я и не смог найти подставу с первого раза.

Проблема может решиться достаточно легко – нужно синхронизировать сервер с локальной копией сайта и удалить на сервере все лишние файлы и заменить все измененные. Идеального решения я не нашел, но есть приближенные к идеалу.



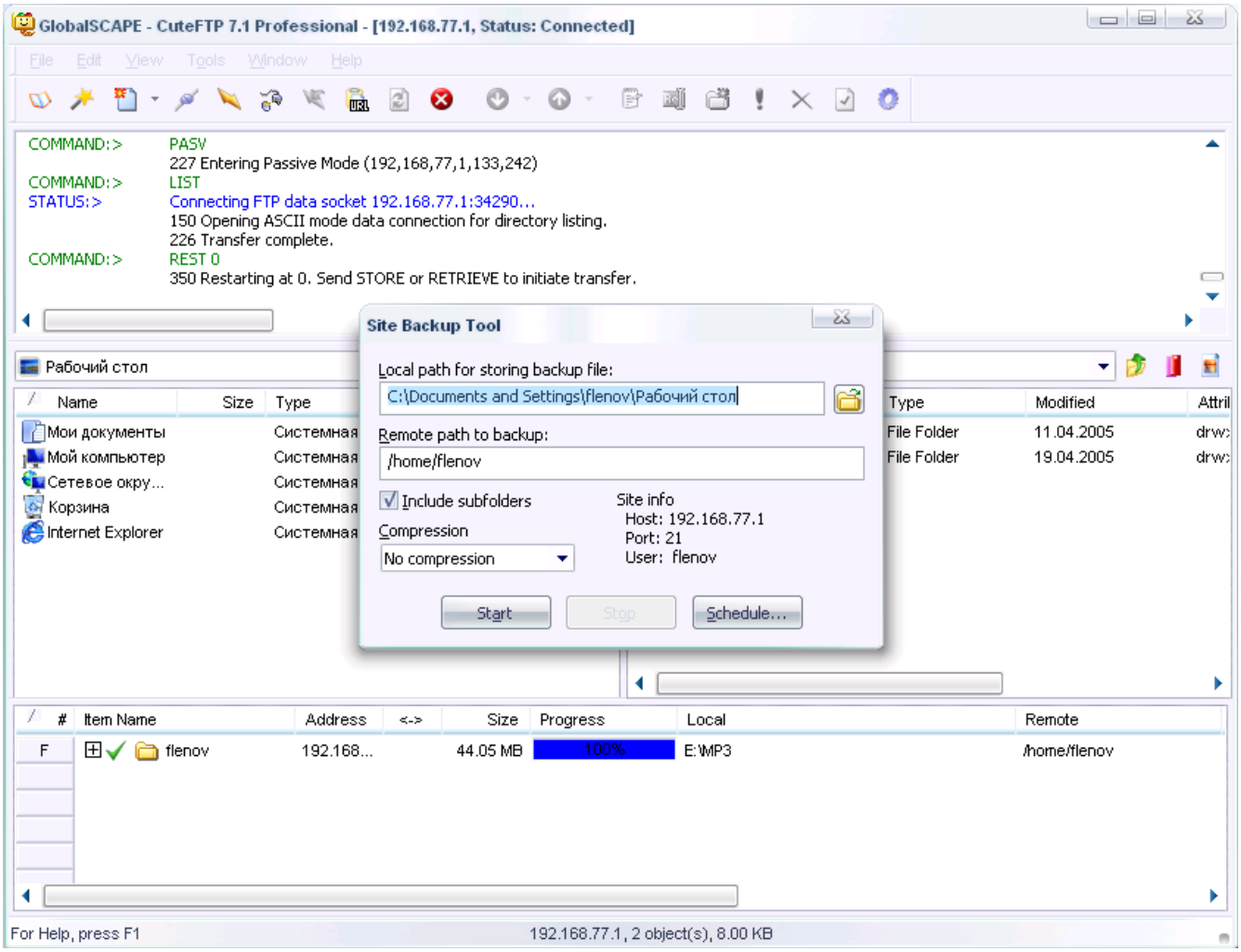
Синхронизация с помощью CuteFTP

Синхронизация файлов

Самым близким к идеалу будет использование легендарного FTP клиента CuteFTP (http://www.globalscape.com/). Запускаем программу, соединяемся с сервером и выбираем пункт меню Tools/Folder tools/Synchronize Folders. Перед тобой появиться окно, в котором нужно указать локальную папку, где находиться копия сайта и удаленную папку сервера. Чтобы синхронизировать не только одну удаленную папку, а все сразу, необходимо выбрать

корневую директорию сайта и поставить галочку в Include subfolders.

И еще, не забудьте в выпадающем списке Please select the direction of synchronization (пожалуйста выберете направление синхронизации) выбрать пункт Mirror local (Make the remote side just like the local), что на нашем родном звучит примерно так: Локальное зеркало (Сделать удаленную сторону точно такой же, как и локальная). Если нажать на кнопку Advanced, то перед нами появляются дополнительные настройки. Здесь обязательно должна быть галочка напротив When one way mirroring, delete destination file if in does not exists in source folder (При односторонней синхронизации удалять файлы на приемнике, если они не существуют на источнике). Это значит, что если локально какой-то файл не существует, то он будет удален на сервере, потому что он создавался не вами и скорей всего содержит заразу.



Резервирование файлов с помощью CuteFTP

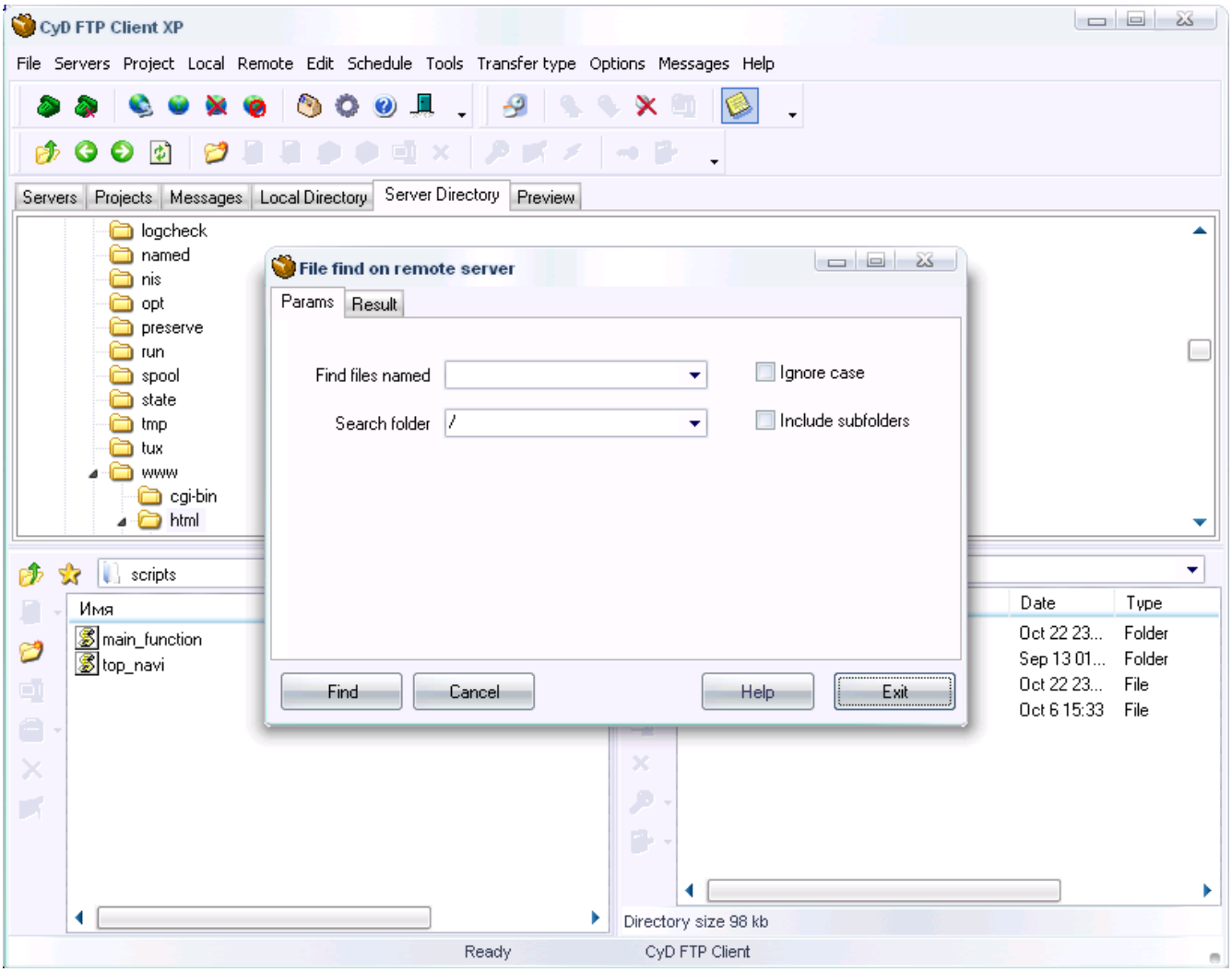
Единственный недостаток такого решения – директория на локальном компьютере должна содержать точную копию сайта. Это не всегда удобно. Иногда бывают папки, которые занимают много места и держать их локально просто влом. Например, у меня это папки Download и Files, где полно мусора. Если нет полной копии на локальном компьютере, то это будет серьезная проблема во время синхронизации. Просто удалиться то, что нужно, и этим мы только рассмешим хакера и посетителей. А ведь в zip архивах шелл врят ли будет жить, да и лишний php файл среди архивов с другим расширением будет сразу виден, как обезьяна среди людей.

Есть еще один недостаток – CuteFTP иногда ошибается. Дело в том, что если сервер находиться в Москве, а ты на Чукотке, то часы будут показывать разное время и, закачав файлы на сервер, дата будет отличаться от даты локального файла. Проследить это CuteFTP может, но не всегда, поэтому и возникают проблемы в синхронизации.

Резервирование сайта

Все необходимые файлы, особенно сценарии чаще всего есть не только на сервере, но и на локальном компьютере, потому что все сценарии пишутся локально, и только потом загружаются на сервер. Но бывают случаи, когда над файлами работают несколько человек или целая команда и у каждого есть своя версия. В этом случае, кто-то один должен хранить полную копию и резервировать все файлы к себе. Для этого опять же можно воспользоваться программой CuteFTP.

Для резервирования файлов выбираем меню Tools/Folder tools/Backup remote folders. Оригинально тут то, что можно сразу же сжать все файлы и директории (включая подкаталоги) в архив.

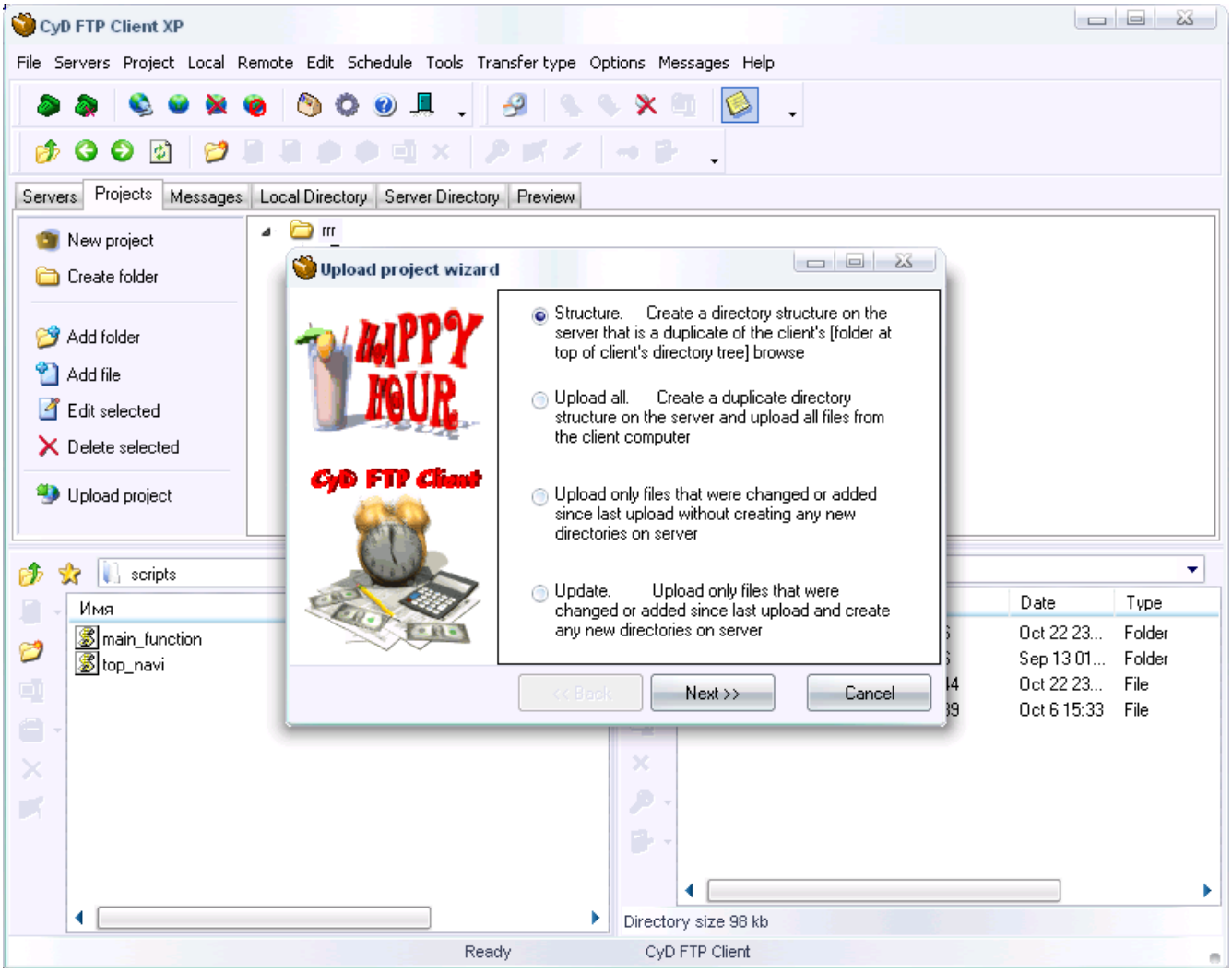


Поиск заразных файлов

Перезагрузка файлов

Самый надежный способ восстановить работоспособность сайта и при этом удалить все последствия вторжения – удалить содержимое сайта и перекачать его заново. Если структура файлов на сервере хорошо продумана, то таким способом можно сэкономить трафик. Например, все архивы должны лежать в отдельной директории и там не должно быть никаких сценариев или WEB страниц. То же самое и с картинками. В этом случае, достаточно удалить и загрузить только сценарии, а директории с архивами и картинками только просмотреть на наличие опасности.

Для решения этой проблемы я использую свой CyD FTP Client XP (http://www.cydsoft.com). Сначала я удаляю все директории с сценариями и потом запускаю поиск на сервере на предмет наличия htm, php, dat и т.д. Если что-то подобное найдено в не положенном месте, то не думая удаляю. Для поиска выбираем меню Edit/File find on remote server.



Загрузка проекта с помощью CyD FTP Client XP

После чистки сервера загружаем заново все необходимые и корректные файлы из локальной копии. Для этого используем проекты. Вверху окна выбираем закладку Project и создаем здесь проект. В проекте создаем структуру сайта, при этом, на локальном компьютере файлы

могут находиться в любом порядке. Главное, чтобы в дереве проекта они находились, каждый на своем месте. После этого выбираем пункт меню Project/Upload project и запускается мастер загрузки проекта. Следуя инструкциям мастера можно быстро восстановить/обновить содержимое сервера. Это намного удобнее синхронизации, когда на локальном компьютере структура папок обязательно должна совпадать со структурой сервера.

Проекты CyD FTP Client XP очень удобны при обновлении сайта. Достаточно запустить мастер загрузки и попросить обновить сервер. Во время выполнения этой операции не происходит сравнения файлов по дате, вместо этого, программа сравнивает даты из собственного журнала (какая версия файла загружалась последней и какая сейчас находиться на локальном компьютере), что исключает проблему не совпадения дат.

Хранение резервной копии

Недостаток дампа MySQL – это текстовый файл, который занимает слишком много места. Дисковое пространство действительно улетучивается слишком много, потому что mysqldump не экономит место и вставляет кучу пробелов, пустых строк и комментариев. Намного эффективнее будет сжать этот файл, например, с помощью gzip, иначе это будет слишком шикарное расточительство дискового пространства, да и качать к себе не сжатый дамп слишком накладно для трафика.

Чтобы заархивировать файл необходимо выполнить две команды:

```
tar cf backup.tar /home/dump.sql

gzip -9 backup.tar
```

Чтобы разархивировать такой файл, выполняем команду:

```
tar xfz /home/backup.tar.gz
```

Как видишь, все очень просто, а результат – реальная экономия.

Защита резервной копии

Нет смысла защищать систему, если резервные копии беспорядочно лежат у вас по всему диску. Очень часто в базе данных находятся очень важные данные. Даже просто структура базы уже является секретной, потому что позволяет хакеру узнать много лишнего, чего ему не положено знать, ведь зная имена объектов, упрощается реализация атаки SQL-Injection. Однажды я видел, как секретные данные с хорошо защищенного сервера каждый час копировались на простой компьютер пользователя, на котором все настройки были установлены по умолчанию. Такую систему хакер взломает за пять минут.

К защите резервных копий нужно подходить со всей ответственностью. Самый простой вариант поместить их в сейф, а самый лучший – это вытащить из сейфа и закопать где-нибудь в Сибири на глубину 100 метров. Тогда ни один хакер не найдет ваших данных. Но каждый раз ездить в Сибирь накладно, поэтому лучшим вариантом будет перед записью резервных копий на носитель зашифровать файл. Напоминаю (или сообщаю для тех, кто не знал), что зашифровать файл с резервной копией backup.tar.gz с помощью пакета OpenSSH можно следующей командой:

```
/usr/bin/openssl des –in /home/backup.tar.gz –out /home/backup.sec
```

В ответ на это будет создан файл backup.sec. Именно его и надо записывать на носитель для

долгосрочного хранения. Только не забудьте удалить потом из компьютера файлы backup.tar.gz и backup.sec.

При восстановлении, файл сначала необходимо расшифровать:

```
/usr/bin/openssl des –d –in /home/backup.sec \

–out /home/backup.tar.gz
```

После этого уже можно разархивировать все файлы на свои места.

Backup Complete

Надеемся, что мы смогли убедить тебя в необходимости создания резервных копий всего, что храниться на сервере, и ты уже выбрал лучший вариант резервирования. Осталось только определиться с планом резервирования (когда будет производиться создание дампа базы и резервирование файлов) и начать действовать. Главное, никогда не опускать руки. Беда всегда приходит неожиданно, когда ты забудешь создать резервную копию, а если под рукой есть все необходимое для восстановления работоспособности, то это уже не беда.

Помни, хакерство – это образ мышления, образ жизни и постоянные исследования. Удачи в твоих дальнейших исследованиях.

Пример создания дампа базы на PHP

```
$db='имя базы';


// Подключаемся к базе

@mysql_select_db($db);


// Получаем список таблиц

$res = mysql_query("SHOW TABLES FROM $db");

if (mysql_num_rows($res)==0)

{ // Если база не содержит таблиц, то дамп делать нельзя, иначе будет зависон

 print("База данных пустая. Дамп невозможен");

}

else

{

 while ($row = mysql_fetch_row($res))
```

```
{
 $tabs[] .= $row[0];
}
// Запуск цикла, выполняющего дамп
print("# Дамп базы данных: $db<BR><BR>");
foreach($tabs as $tab)
{
 // Отображаем SQL код удаления базы
 print("# Drop table ".$tab." <BR> DROP TABLE IF EXISTS '".$tab."';<BR><BR>");

 // Определяем SQL код создания базы
 $res = mysql_query("SHOW CREATE TABLE `".$tab."`") or die(mysql_error());
 $row = mysql_fetch_row($res);
 // Отображаем SQL код создания базы
 print("# Create table ".$tab." <BR> ".$row[1].";<BR><BR># Data:<BR>");

 // Получаем данные таблицы
 $res = mysql_query("SELECT * FROM `$tab`");
 if (mysql_num_rows($res) > 0)
 {
  while ($row = mysql_fetch_assoc($res))
  {
    // Отображаем SQL код добавления строки
    $keys = implode("`, `", array_keys($row));
    $values = array_values($row);
    foreach($values as $k=>$v) {$values[$k] = addslashes($v);}
    $values = implode("', '", $values);
    print("INSERT INTO `$tab`(`".$keys."`) VALUES ('".$values."');<BR>");
  }
```

```
  }
  print("# ====================================<BR><BR>");
 }
}
```

Советы

Если сайт пострадал от хакеров, то перед восстановлением необходимо подумать о том, чтобы найти способ проникновения хакеров в свои владения и прикрыть ворота. Если этого не сделать, то после восстановления сайт снова может пострадать. Но это уже совсем другая история.

Резервировать нужно не только рабочую базу данных, но и системную mysql, в которой хранятся все права и описания объектов сервера. Если есть возможность, то рекомендую резервировать mysql после каждого внесения изменений в структуру данных и права доступа.

Не перезаписывай резервные копии, а храни их за каждую дату в отдельном архиве. Так ты сможешь вернуть на несколько дней назад и посмотреть, что было с данными.

Готовься заранее к самому худшему. Никогда не знаешь, что может произойти, а по закону подлости произойдет то, к чему не готовишься.