Linux入侵踪迹隐藏攻略Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/645/2021_2022_Linux_E5_85_A5_E4_BE_c103_645356.htm 0．前言：被警察叔叔请去喝茶时间很痛苦的事情，各位道长如果功力不够又喜欢出风头的想必都有过这样的“待遇”。如何使自己在系统中隐藏的更深，是我们必须掌握的基本功。当然，如果管理员真的想搞你而他的功力又足够足的话，相信没什么人能够真正的“踏雪无痕”。Forensic与Anti- Forensic，说到底只是你和管理员之间的技术间较量而已。貌似很少有专门说这个的文章，大部分就是下载个日志擦除的软件，然后运行下就可以了，对小站可以，但对方如果是经验丰富的管理员呢？我们该如何应对？我在这里只介绍unix-like system下的，至于windows或者其他什么系统下的，欢迎各位道友补充。 1. 最小化你的日志 P.S 访问目标前用跳板我就不废话了，你是VPN也好3389也罢，ssh中转，代理都行。总之记住一点直接连接攻击目标是愚蠢的 1.1 shell使用问题 目前linux下大多数的shell都是采用bash或者其他的什么shell 通过输入输出重定向来实现与服务器的交互的，当我们使用ssh 或者telnet之类的登录的时候，我们的命令都会被记录在shell 的history文件下面。举例来说bash会在当前目录下面.bash_history文件里记录下你此次登陆操作的命令，如果你拿这台机器当跳板的话，或者扫描其他机器，你的命令都会被记录下来哦。呵呵，所以我们登录的第一件事就是执行如下命令： unset HISTORY HISTFILE HISTSAVE HISTZONE HISTORY HISTLOG. export HISTFILE=/dev/null. export HISTSIZE=0. export

HISTFILESIZE=0 当然不同的shell写法可能不同，像有的set设置环境变量什么的。大家根据自己的shell自行修改。记住：从 webshell弹回的shell也会记录你的操作，值得庆幸的是现在很多弹shell的脚本都预先unset 环境变量。 我们还需要记住的是在登录的时候出现在登录窗口的一些信息，比如该用户在什么时候从哪个IP登录进来的等等，这在我们后面的用于日志清除与修改的时候要用到。 如图： 作为跳板的时候，我们有可能需要用本机的ssh去访问别的机器，但是别的机器的公钥呢？总不能放在当前用户的目录下吧？当然你可以事后删除，但多一事不如少一事，你说对么？ ssh -o UserKnownHostsFile=/dev/null -T user@host /bin/bash i 就可以了，但在这样运行某些命令的时候可能会有提示，说你的stdin不是个terminal，这里可以这样解决： python -c import pty. pty.spawn("/bin/sh") 或者自己再建立个ttyshell。 1.2 webshell的选择问题 可能各位道友的日常生活中最主要目标瞄向了webserver 。现在的web也是大多数入侵的一个突破口。Linux下用的最多的就是apache服务器了，当我们发觉一个服务器的漏洞时候很可能要上传一个webshell来进行对服务器文件进一步的操作和信息的搜集，部分webshell也提供了反弹shell的功能。如何能够在apache服务器的日志文件中留下最小的记录也是需要深究的。这种情况通常发生在没能够获得足够的权限来清除apache日志。如果能够 root了，则可以将重点放在第二节日志清除上。通常，日志只记录GET的信息，比如你的注入，你采用了那种方式提交数据等等。如果我们的webshell采用的多是GET方式交互的话，就很容易在httpd的access_log中留下很多日志。这些以后都会被作为证据所采

纳的。 Phpspy是个很好的选择，作者也注意掉了这点，取消了GET方式的交互，再给webshell起一个比较迷惑的名字，这样我们与webshell的交流就更加隐秘。 2.日志的清除与改写 日志清除与改写，俗称擦PP，这是个很重要的过程，日志记录了你对目标机器的操作记录，大部分的入侵者查找都是通过日志来确定的，因此，我们需要对日志文件进行操作。对日志操作有这么个说法，能修改的就不清除，这样才能最小的减少管理员的怀疑。Linux下的大多数文件是以文本方式，或者以简单的结构体方式存入文件的，这就为我们修改某个日志记录里的具体内容提供了前提条件。 需要注意的一点是，我们需要先看看日志的存放位置，有的管理员会修改日志保存的位置，一般来说，我们可以查看/etc/syslog.conf 来获得log文件存放的位置。但要注意的是，有的管理员(及其负责)会重新编译syslogd文件来重新指定log存放的位置，怎么办？在这种情况下可以用strings来看下/sbin/syslogd这个文件，这种管理员我只在书里看到过，至少我没遇到过: P。这个配置文件里面记录了系统存放某些log的目录，如secure文件等。下面我们就会根据这个文件来清理和修改日志。 现在可以在网上公开获得的日志清除程序代码很粗糙，我曾经看到过最夸张的清日志的代码像这样: rm -rf /var/log/lastlog . rm -rf /var/log/telnetd . rm -rf /var/run/utmp . rm -rf /var/log/secure . rm -rf /root/.ksh_history . rm -rf /root/.bash_history . rm -rf /root/.bash_logut . rm -rf /var/log/wtmp . rm -rf /etc/wtmp . rm -rf /var/run/utmp . rm -rf /etc/utmp . rm -rf /var/log . rm -rf /var/adm . rm -rf /var/apache/log . rm -rf /var/apache/logs . rm -rf /usr/local/apache/log . rm -rf /usr/local/apache/logs . rm -rf

/var/log/acct . rm -rf /var/log/xferlog . rm -rf /var/log/messages . rm -rf /var/log/proftpd/xferlog.legacy . rm -rf /var/log/proftpd.access_log . rm -rf /var/log/proftpd.xferlog . rm -rf /var/log/httpd/error_log . rm -rf /var/log/httpd/access_log . rm -rf /etc/httpd/logs/access_log . rm -rf /etc/httpd/logs/error_log .rm -rf /var/log/news/suck.notice . rm -rf /var/spool/tmp . rm -rf /var/spool/errors . rm -rf /var/spool/logs . rm -rf /var/spool/locks . rm -rf /usr/local/www/logs/thttpd_log . rm -rf /var/log/thttpd_log . rm -rf /var/log/ncftpd/misclog.txt . rm -rf /var/log/ncftpd.errs . rm -rf /var/log/auth . rm -rf /root/.bash_history . touch /root/.bash_history . history r 整个一rm集合，要是服务器跑了很长时间，积累了很多日志。你这样一删除，的，你帮他省事了，他也省事，一眼就看出有人进来了。 