Linux手工入侵排查思路

原創 Bypass Bypass 今天

當Linux主機發生安全事件需要進行入侵排查時，一般可以使用常見的shell命令，通過分析主機的異常現象、進程端口、啟動方式、可疑文件和日誌記錄等信息以確認主機是否被入侵。

在這裡，結合工作中Linux安全事件分析處理辦法，總結了Linux手工入侵排查過程中的分析方法。

01、檢查系統賬號

從攻擊者的角度來說，入侵者在入侵成功後，往往會留下後門以便再次訪問被入侵的系統，而創建系統賬號是一種比較常見的後門方式。在做入侵排查的時候，用戶配置文件/etc/passwd和密碼配置文件/etc/shadow是需要去重點關注的地方。

（1）查詢特權用戶特權用戶(uid 為0)

```
awk -F: '$3==0{print $1}' /etc/passwd
```

（2）查詢可以遠程登錄的帳號信息

```
awk '/\$1|\$6/{print $1}' /etc/shadow
```

（3）除root帳號外，其他帳號是否存在sudo權限。如非管理需要，普通帳號應刪除sudo權限

```
more /etc/sudoers | grep -v "^#\|^$" | grep "ALL=(ALL)"
```

（4）禁用或删除多餘及可疑的帳號

```
usermod -L user     禁用帐号，帐号无法登录，/etc/shadow第二栏为!开头
userdel user        删除user用户
userdel -r user     将删除user用户，并且将/home目录下的user目录一并删除
```

（5）當前登錄當前系統的用戶信息

```
who     查看当前登录用户（tty本地登陆  pts远程登录）
w       查看系统信息，想知道某一时刻用户的行为
uptime  查看登陆多久、多少用户，负载
```

02、檢查異常端口

（1）使用netstat 網絡連接命令，分析可疑端口、IP、PID等信息。

```
netstat -antlp|more
```

（2）如發現異常的網絡連接需要持續觀察，可抓包分析

```
tcpdump -c 10 -q   //精简模式显示 10个包
```

03、檢查可疑進程

（1）使用ps命令列出系統中當前運行的那些進程，分析異常的進程名、PID，可疑的命令行等。

```
ps aux / ps -ef
```

（2）通過top命令顯示系統中各個進程的資源佔用狀況，如發現資源佔用過高

```
top
```

（3）如發現異常，可使用一下命令進一步排查：

```
查看该进程启动的完整命令行: ps eho command -p $PID
查看该进程启动时候所在的目录: readlink /proc/$PID/cwd
查看下pid所对应的进程文件路径：ls -l /proc/$PID/exe
查看该进程启动时的完整环境变量: strings -f /proc/1461/environ | cut -f2 -d ''
列出该进程所打开的所有文件: lsof -p $PID
```

04、檢查系統服務

Linux系統服務管理，CentOS7使用systemd控制 CentOS6之前使用chkconfig控制。

（1）對於systemd服務管理器來說，可以通過下述方式查看開機自啟的服務:

```
systemctl list-unit-files --type=service | grep "enabled"
```

（2）chkconfig就是CentOS6以前用来控制系统服务的工具，查看服务自启动状态：

```
chkconfig  --list
chkconfig --list | grep "3:on\|5:on"
```

05、检查开机启动项

（1）检查启动项脚本

```
more /etc/rc.local /etc/rc.d/rc[0~6].d ls -l /etc/rc.d/rc3.d/
```

（2）例子:当我们需要开机启动自己的脚本时，只需要将可执行脚本丢在/etc/init.d目录下，然后在/etc/rc.d/rc*.d中建立软链接即可

```
ln -s /etc/init.d/sshd /etc/rc.d/rc3.d/S100ssh
```

此处sshd是具体服务的脚本文件，S100ssh是其软链接，S开头代表加载时自启动；如果是K开头的脚本文件，代表运行级别加载时需要关闭的。

06、检查计划任务

利用计划任务进行权限维持，可作为一种持久性机制被入侵者利用。检查异常的计划任务，需要重点关注以下目录中是否存在恶意脚本。

```
/var/spool/cron/*
/etc/crontab
/etc/cron.d/*
/etc/cron.daily/*
/etc/cron.hourly/*
/etc/cron.monthly/*
/etc/cron.weekly/
/etc/anacrontab
/var/spool/anacron/*
```

07、检查异常文件

1、查看敏感目录，如/tmp目录下的文件，同时注意隐藏文件夹，以“..”为名的文件夹具有隐藏属性

2、得到发现WEBSHELL、远控木马的创建时间，如何找出同一时间范围内创建的文件?

```
可以使用find命令来查找，如 find /opt -iname "*" -atime 1 -type f 找出 /opt 下一天前访问过的文件
```

3、针对可疑文件可以使用stat进行创建修改时间。

08、检查历史命令

一般而言，入侵者获取shell之后，会执行一些系统命令从而在主机上留下痕迹，我们可以通过history命令查询shell命令的执行历史。

（1）查询某个用户在系统上执行了什么命令

```
使用root用户登录系统，检查/home目录下的用户主目录的.bash_history文件
```

（2）默认情况下，系统可以保存1000条的历史命令，并不记录命令执行的时间，根据需要进行安全加固。

```
a）保存1万条命令
sed -i 's/^HISTSIZE=1000/HISTSIZE=10000/g' /etc/profile
b）在/etc/profile的文件尾部添加如下行数配置信息：
######jiagu history xianshi#########
USER_IP=`who -u am i 2>/dev/null | awk '{print $NF}' | sed -e 's/[()]//g'`
if [ "$USER_IP" = "" ]
then
USER_IP=`hostname`
fi
export HISTTIMEFORMAT="%F %T $USER_IP `whoami` "
shopt -s histappend
export PROMPT_COMMAND="history -a"
######### jiagu history xianshi ##########
c）source /etc/profile让配置生效
```

09、检查系统日志

在Linux上一般跟系统相关的日志默认都会放到/var/log下面，若是一旦出现问题，用户就可以通过查看日志来迅速定位，及时解决问题。常用日志文件如下：

```
/var/log/btmp：记录错误登录日志，这个文件是二进制文件，不能直接vi查看，而要使用lastb命令查看。
/var/log/lastlog：记录系统中所有用户最后一次登录时间的日志，这个文件是二进制文件，不能直接vi，而要使用lastlog命令查看。
/var/log/wtmp：永久记录所有用户的登录、注销信息，同时记录系统的启动、重启、关机事件。同样这个文件也是一个二进制文件，不能直接vi，而
/var/log/utmp：记录当前已经登录的用户信息，这个文件会随着用户的登录和注销不断变化，只记录当前登录用户的信息。同样这个文件不能直接v
/var/log/secure：记录验证和授权方面的信息，只要涉及账号和密码的程序都会记录，比如SSH登录，su切换用户，sudo授权，甚至添加用户和修改
```

一般，我们需要重点去关注secure安全日志，检查系统错误登陆日志，统计IP重试次数，成功登录的时间、用户名和ip，确认账号是否存在暴力破解或异常登录的情况。

```
1、定位有多少IP在爆破主机的root帐号：
grep "Failed password for root" /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr | more

定位有哪些IP在爆破：
grep "Failed password" /var/log/secure|grep -E -o "(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|

爆破用户名字典是什么？
 grep "Failed password" /var/log/secure|perl -e 'while($_=<>){ /for(.*?) from/; print "$1\n";}'|uniq -c|sort -nr

2、登录成功的IP有哪些：
grep "Accepted " /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr | more
```

```
12
13 登录成功的日期、用户名、IP：
14 grep "Accepted " /var/log/secure | awk '{print $1,$2,$3,$9,$11}'
```