**Linux主机安全基线操作手册**

**目录**

# 安全加固

# （需要人工操作）

## 1.1 使用Centos6.8 作为服务器的指定操作系统，使用64位操作系统

|  |  |
| --- | --- |
| **操作步骤** | N/A |
| **备注** |  |

## 1.2系统安装完成后，必须使用公司的升级源（yum server）升级到最新的安全稳定版，包含最新的安全补丁。

|  |  |
| --- | --- |
| **操作步骤** | 升级到3.2.29版本 |
| **备注** |  |

## 1.3只允许使用公司的yum服务器作为更新源，同时打开PGP校验，并使用在线RHEL的 pgp key进行校验。

|  |  |
| --- | --- |
| **操作步骤** | 1. 确认系统已经安装GPG公钥   **# rpm -q --queryformat "%{SUMMARY}\n" gpg-pubkey**  如返回如下字符则表示已经安装GPG公钥  **gpg(Red Hat, Inc. (release key <security@redhat.com>)**  **如无返回则需安装GPG公钥**  **#yum install gnupg**   1. 确认包签名检查已经激活，检查/etc/yum.conf文件，确认[main]标签下存在如下字段   **gpgcheck=1**   1. 检查/etc/yum.repos.d目录下不会出现gpgcheck=0字段   **#find \* | xargs grep "gpgcheck=0"** |
| **备注** | 使用数字证书检查软件来源的安全性 |

## 1.4．系统只安装公司认可的应用程序，禁止安装编译环境（gcc，automake等），远程管理软件、代理服务器软件（http，socks ），VPN软件，抓包软件tcpdump ethereal wireshark等。(限制只有root用户才可使用)。

|  |  |
| --- | --- |
| 操作步骤 | 卸载 gcc|automake|ethereal|wireshark |
| 备注 |  |

## 1.5.grub使用MD5密码进行保护。

|  |  |
| --- | --- |
| 操作步骤 | 1. 生成grub md5密码   #grub-md5-crypt  Password: 输入明文密码  Retype password:再次输入明文密码  此时会产生密文输出如，$1$wTCWa$sblbkYADaNNIG6sMUUWgH1   1. 编辑/etc/grub.conf文件，在首行添加如下上条命令产生的密文   password --md5 $1$wTCWa$sblbkYADaNNIG6sMUUWgH1 |
| **备注** | 保护grub,防止随意进入单用户模式修改root密码 |

## 1.6.设置本地控制台超时退出。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/profile.d/tmout.sh，添加如下语句  TMOUT=600 |
| 备注 |  |

## 1.7.禁止不常用的文件格式、模块

/etc/modprobe.conf 添加以下：

install cramfs /bin/true

install freevxfs /bin/true

install jffs2 /bin/true

install hfs /bin/true

install hfsplus /bin/true

install squashfs /bin/true

install udf /bin/true

install ppp\_generic /bin/true

install pppoe /bin/true

install pppox /bin/true

install slhc /bin/true

install bluetooth /bin/true

install ipv6 /bin/true

install irda /bin/true

install ax25 /bin/true

install x25 /bin/true

install ipx /bin/true

install appletalk /bin/true

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/modprobe.conf，确保存在如下语句  install cramfs /bin/true  install freevxfs /bin/true  install jffs2 /bin/true  install hfs /bin/true  install hfsplus /bin/true  install squashfs /bin/true  install udf /bin/true  install ppp\_generic /bin/true  install pppoe /bin/true  install pppox /bin/true  install slhc /bin/true  install bluetooth /bin/true  install ipv6 /bin/true  install irda /bin/true  install ax25 /bin/true  install x25 /bin/true  install ipx /bin/true  install appletalk /bin/true |
| 备注 |  |

## 1.8.崩溃保护。禁止SUID Core Dumps设置sysctl fs.suid\_dumpable=0。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/sysctl.conf，确保存在如下语句  fs.suid\_dumpable = 0 |
| 备注 |  |

## 1.9.默认启动init级别设置为3。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/inittab，确保存在如下语句  id:3:initdefault: |
| 备注 |  |

## 1.10.禁用ssh 1协议。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/ssh/sshd\_config，确保存在如下语句  Protocol 2 |
| 备注 |  |

## 1.11.设置LogLevel为INFO。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/ssh/sshd\_config，确保存在如下语句  LogLevel INFO |
| 备注 |  |

## 1.12.忽略rhost请求。（除非备份机器需要）

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/ssh/sshd\_config，确保存在如下语句  IgnoreRhosts yes |
| 备注 |  |

## 1.13.禁用TCP转发。（除非备份机器需要）。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/ssh/sshd\_config，确保存在如下语句  AllowTcpForwarding no |
| 备注 |  |

## 1.14.禁止空密码。

|  |  |
| --- | --- |
| 操作步骤 | 编辑/etc/ssh/sshd\_config，确保存在如下语句  PermitEmptyPasswords no |
| 备注 |  |

## 1.15.nginx安全加固

* 1. 约定nginx使用 rpm安装。版本号、安装目录、配置文件需要标准化。
  2. 返回自定义版本号。

|  |  |
| --- | --- |
| 操作步骤 | 使用1.8.0或1.6.2版本 |
| 备注 |  |

## 1.16.php使用rpm。版本号、安装目录、配置文件需要标准化。(5.5.25)

|  |  |
| --- | --- |
| **操作步骤** | **使用5.5.25版本** |
| **备注** |  |

## 1.17.日志级别设置为 E\_ALL，错误日志打到syslog。

|  |  |
| --- | --- |
| **操作步骤** | **编辑php.ini文件，查看是否有如下项，如无，请添加**  **#vi /etc/php.ini**  **error\_reporting = E\_ALL**  **error\_log = syslog** |
| **备注** | 进行日志审计 |

## 1.18.不在页面中显示PHP的错误。

|  |  |
| --- | --- |
| **操作步骤** | **编辑php.ini文件，查看是否有如下项，如无，请添加**  **#vi /etc/php.ini**  **display\_errors = Off** |
| **备注** |  |

## 1.19.session.cookie\_httponly = 1

|  |  |
| --- | --- |
| **操作步骤** | **编辑php.ini文件，查看是否有如下项，如无，请添加**  **#vi /etc/php.ini**  **session.cookie\_httponly = TRUE** |
| **备注** |  |

## 1.20.Iptables服务必须启用。

|  |  |
| --- | --- |
| 操作步骤 | /etc/init.d/iptables start |
| 备注 |  |

## 1.21.系统的shell精简。只保留bash。

|  |  |
| --- | --- |
| **操作步骤** | **删除其它shell** |
| **备注** |  |

## 1.22.Syslog服务必须启动。

|  |  |
| --- | --- |
| **操作步骤** | **#/etc/init.d/rsyslog start** |
| **备注** |  |

## 1.23.所有Syslog发送到log.eformax.com。

|  |  |
| --- | --- |
| **操作步骤** | **vi /etc/rsyslog.conf,确保存在如下语句**  **\\*\.\\*.\*log.eformax.com** |
| **备注** |  |

## 1.24./etc/syslog.conf里添加

kern.\*                                                  /var/log/kern.log

authpriv.\*                                              /var/log/secure

\*.\* @log.eformax.com

|  |  |
| --- | --- |
| 操作步骤 | vi /etc/rsyslog.conf,确保存在如下语句  kern.\*                                                  /var/log/kern.log  authpriv.\*                                              /var/log/secure  \*.\* @log.eformax.com |
| 备注 |  |

## 1.25.设置最小基址 sysctl –w vm.mmap\_min\_addr=4096

|  |  |
| --- | --- |
| 方法 | #sysctl –w vm.mmap\_min\_addr=4096 |
| 备注 | 指定用户进程通过mmap可使用的最小虚拟内存地址，以避免其在低地址空间产生映射导致安全问题（崩溃/挂起等） |

## 1.26.禁用SELINUX

|  |  |
| --- | --- |
| 方法 | Vi /etc/selinux/config，确保存在如下语句  SELINUX=disable' |
| 备注 |  |

## 1.27.去掉多余的suid文件

|  |  |
| --- | --- |
| 方法 | #df --local -P |awk {'if (NR!=1) print $6'} |xargs -I '{}' find '{}' -xdev -type f -perm -4000 –print  查看是否存在未授权文件，如存在，请删除属性 |
| 备注 | 查看是否存在未授权文件 |

## 1.28.对system.map文件进行链接，以便主机入侵检测系统进行检查。

|  |  |
| --- | --- |
| 方法 | #ln -s /boot/System.map-`uname -r` /boot/System.map |
| 备注 |  |

## 1.29.必须启动auditd服务

|  |  |
| --- | --- |
| 方法 | #/etc/init.d/auditd start |
| 备注 | 记录系统安全信息，日志存放在/var/log/audit/audit.log里 |

## 1.30.卸载prelink。

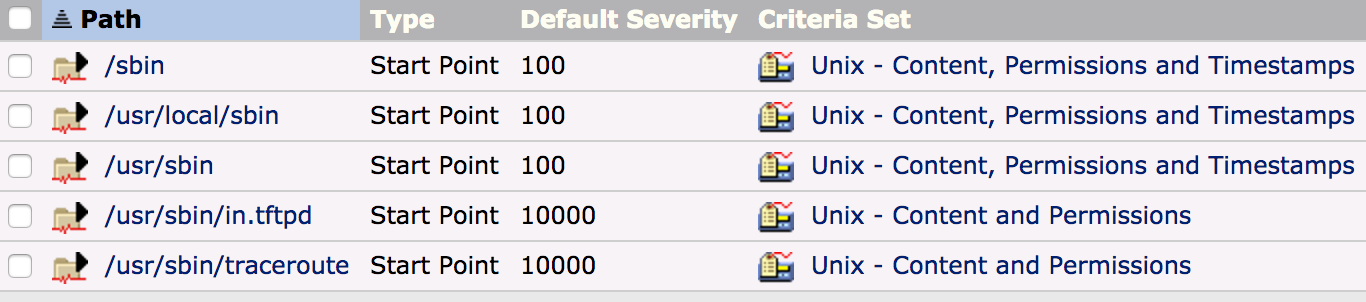
|  |  |
| --- | --- |
| 方法 | 卸载prelink |
| 备注 |  |

## 1.31. 检测openSSH 远程代码执行。

|  |  |
| --- | --- |
| 方法 | if [ "$(rpm -qa openssh-server | grep "7.4p1-1"| grep -v "#")" ]; then echo ok;fi |
| 备注 | Openssh 在 7.4p1-1 以下的存在远程代码执行漏洞。 |

# 安全监控（无需人工操作）

## 2.1 监控管理员命令工具

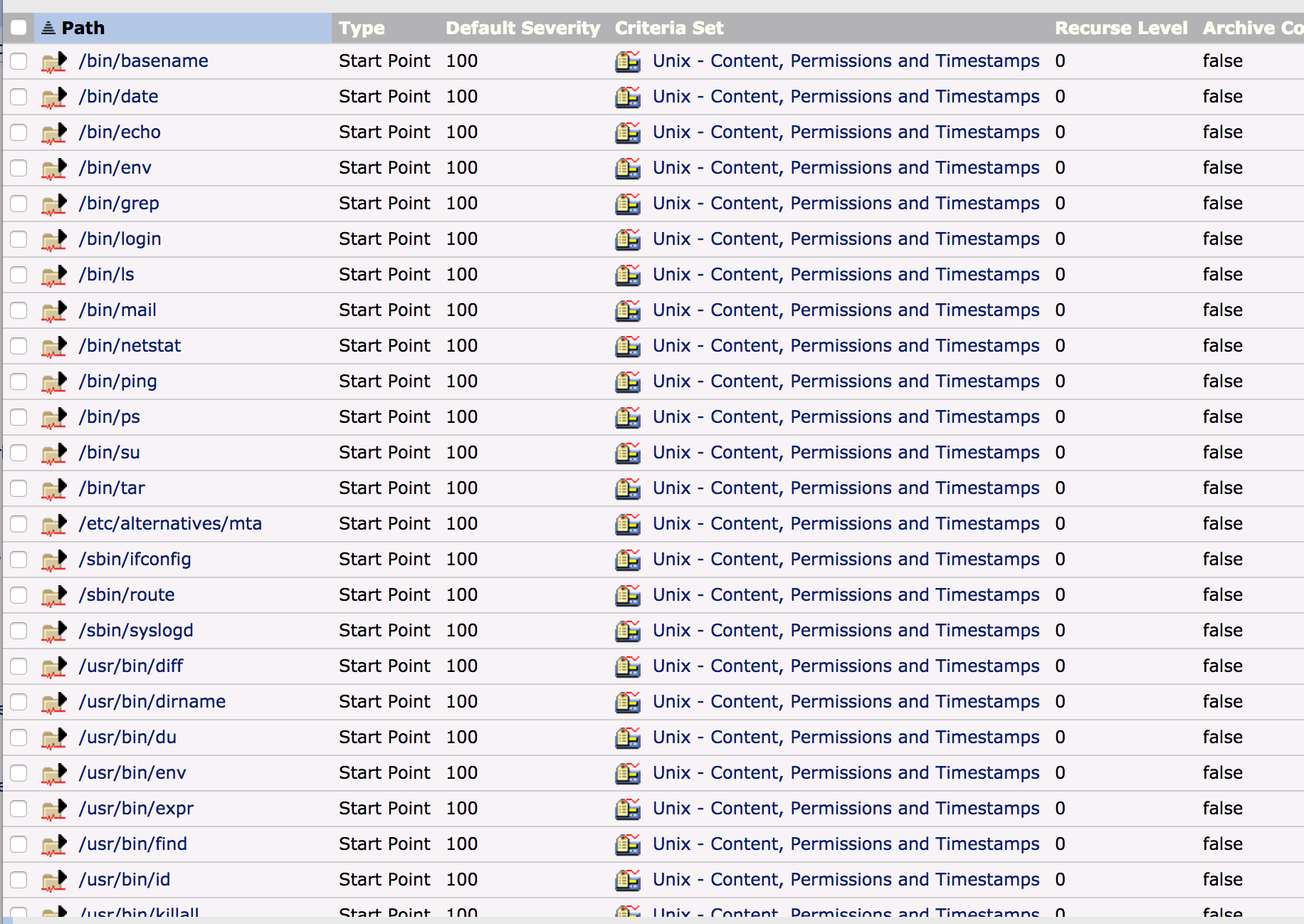
****

**备注：**

**监控系统命令文件，避免黑客替换系统命令，留下后门。**

**并对改动的内容进行监控。**

## 2.2 监控系统提权时常用的命令

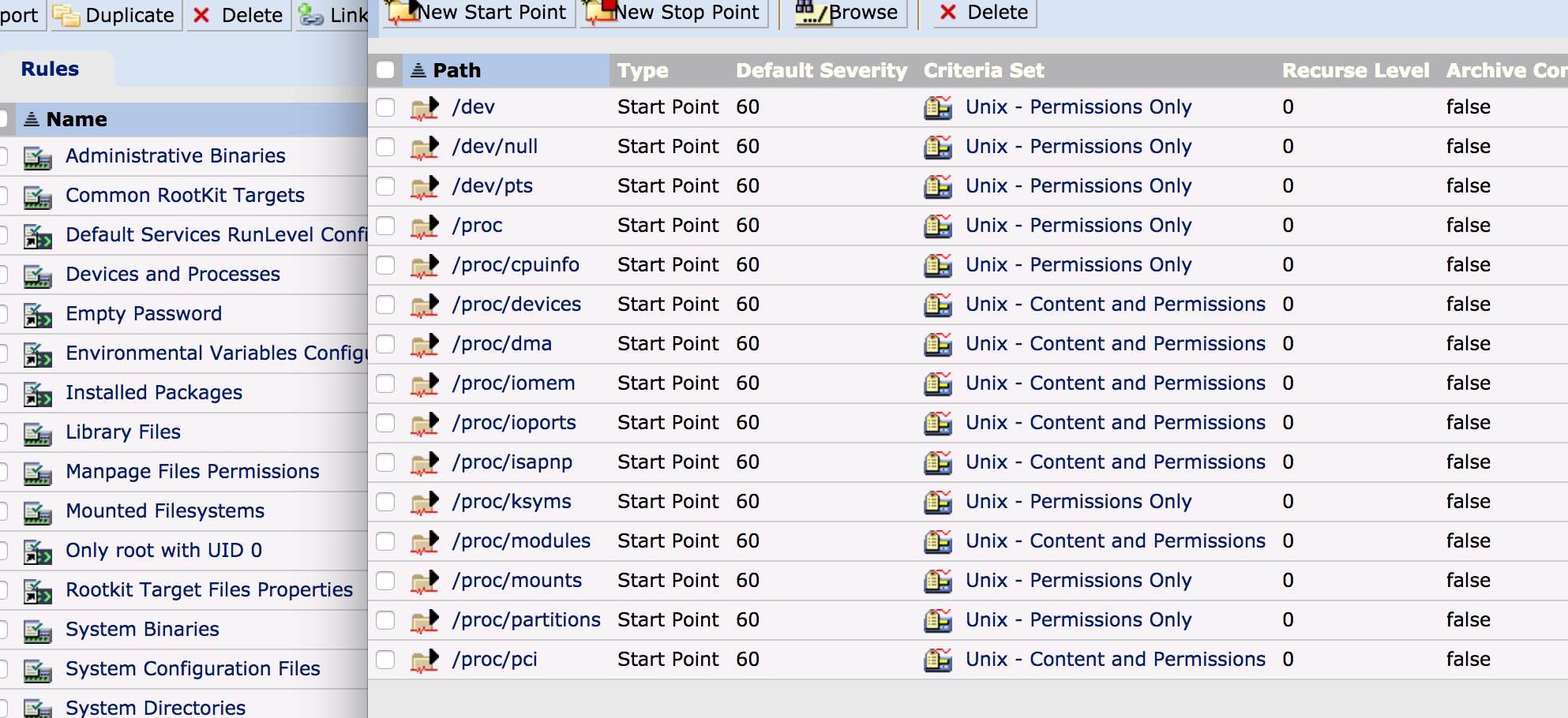
****

**备注： 避免命令被恶意替换，留下后门。**

## 2.3 监控默认启动的系统服务

**相关命令 chkconfig –list**

## 2.4 监控异常设备加载， 与所有进程活动。

****

## 2.5 不允许空密码

**相关命令**

**awk -F: '($2 == "") {print}' /etc/shadow**

## 2.6 监控系统变量改动

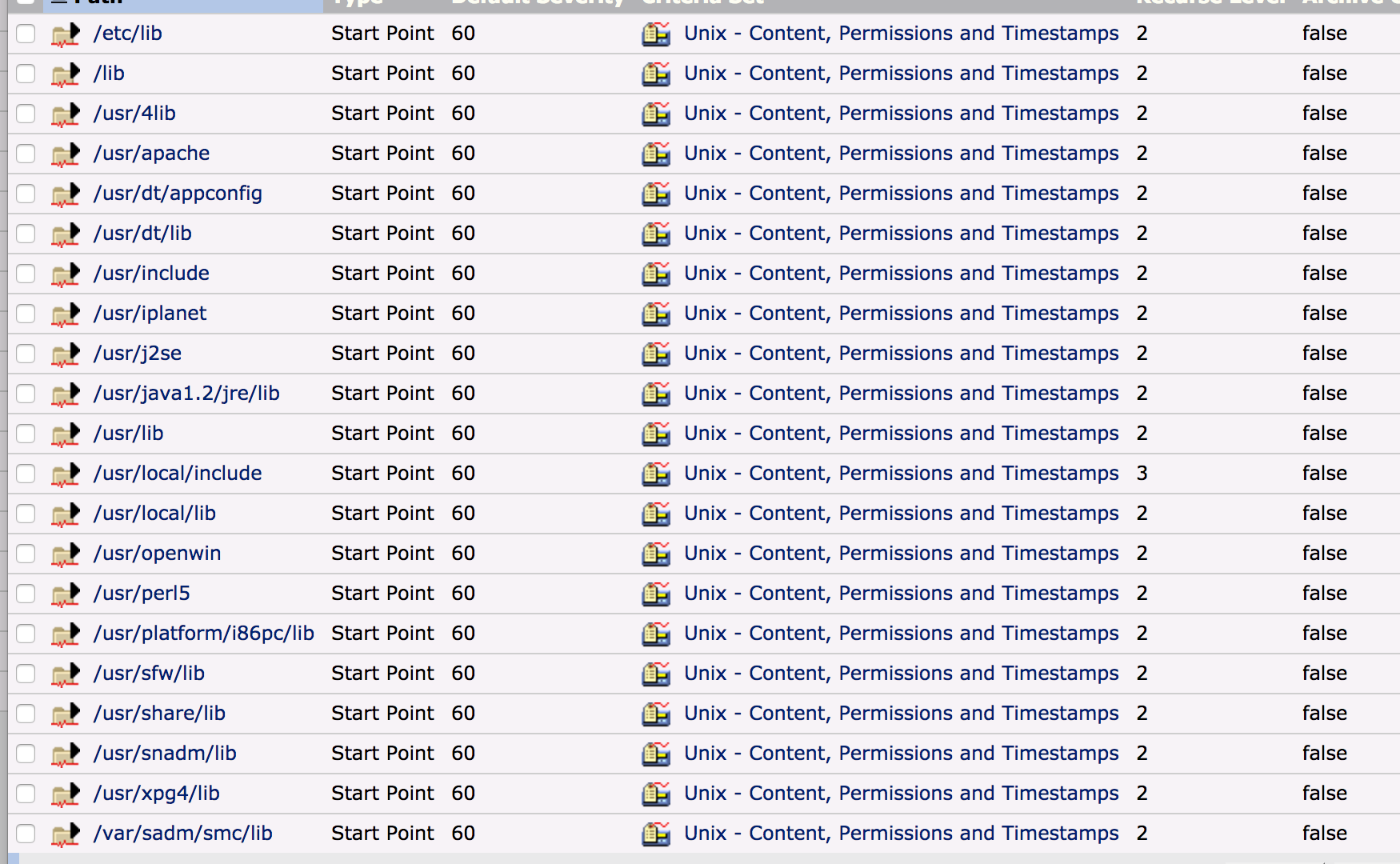
**env | grep -i -e HOME -e PATH -e ROOTPATH -e EDITOR -e SHELL -e TERM**

## 2.7 监控已安装的软件包

**相关命令**

**rpm –qa**

## 2.8 监控系统库文件

****

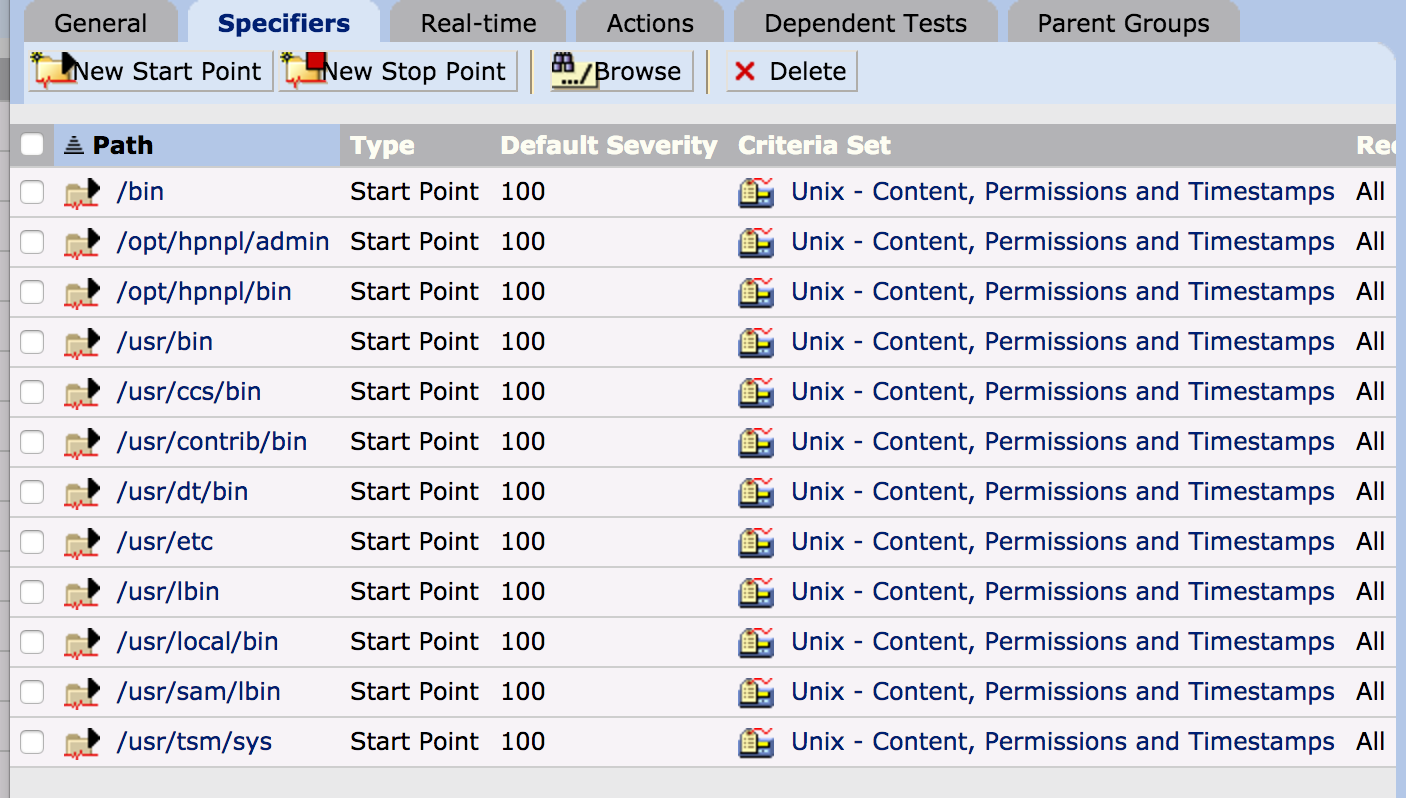
## 2.9 确保只有root 用户的 UID为0

**相关命令：awk -F: '($3 == "0") {print}' /etc/passwd**

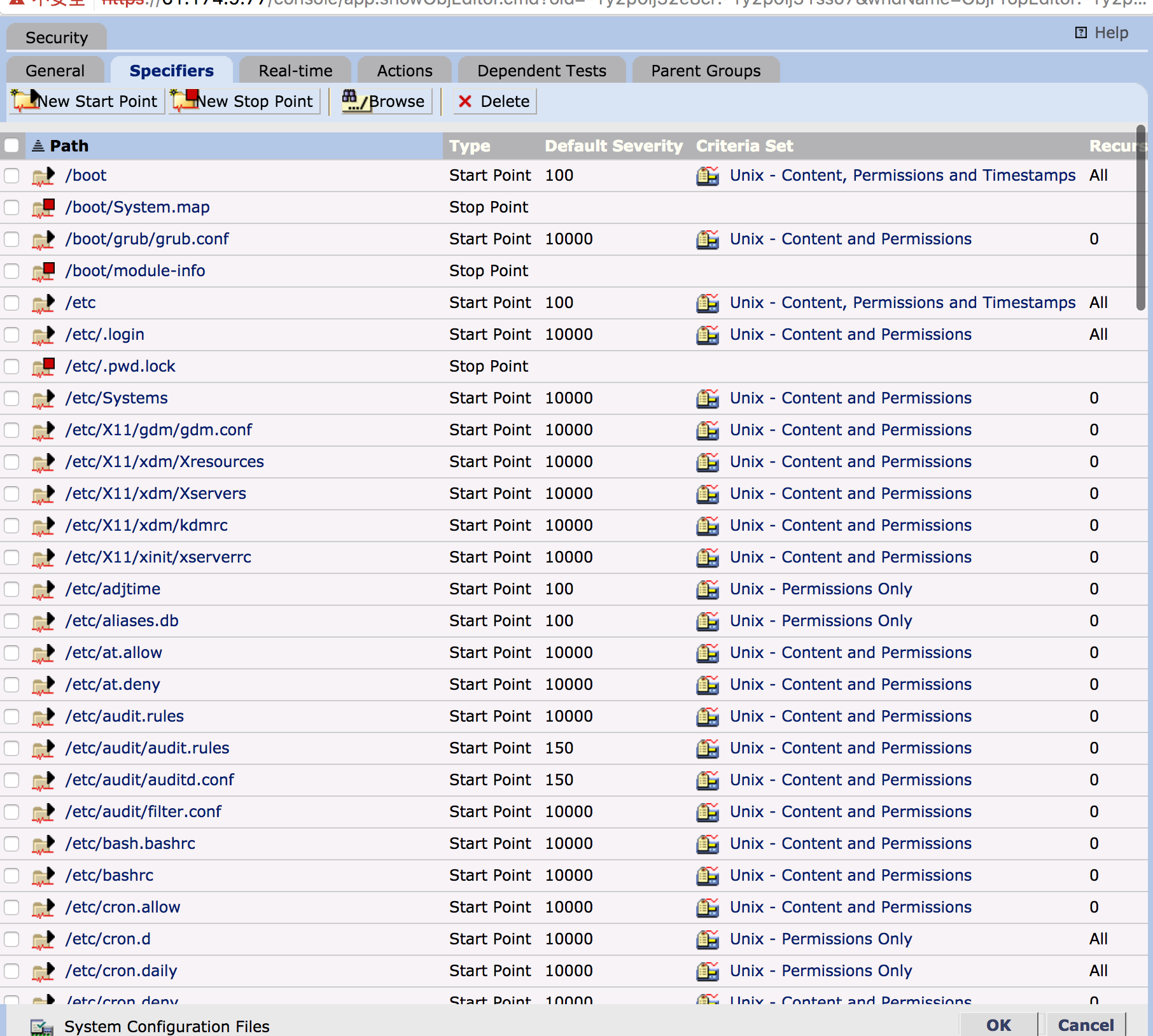
## 2.10 监控提权记录

**lsattr /usr/bin /bin /sbin /usr/sbin 2> /dev/null**

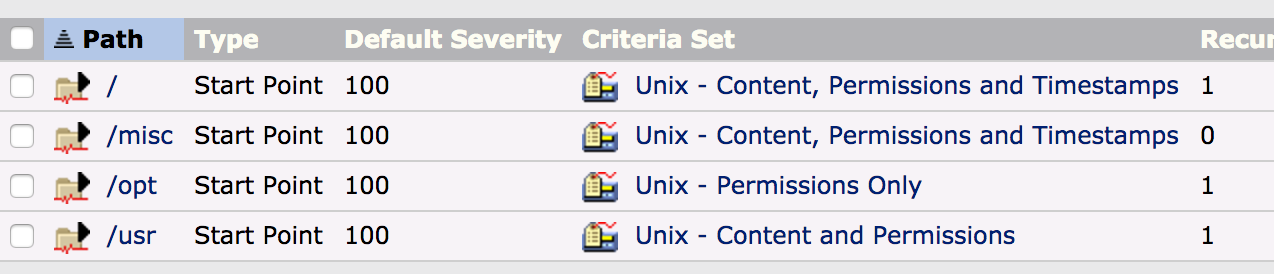
## 2.11 监控系统命令

****

## 2.12 监控系统配置文件

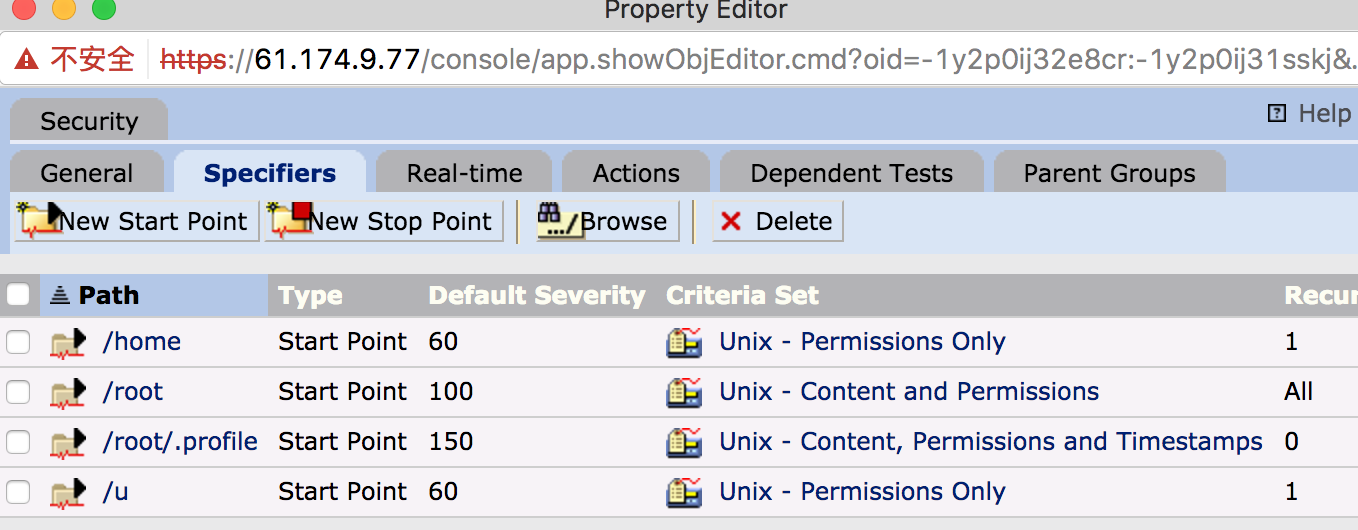
****

## 2.13 监控关键系统目录

****

## 2.14 监控tripwire 配置文件

## 2.15 监控Root用户目录与用户目录权限

****

## 2.16 监控关键系统文件权限变更

****

## 2.17 审计root 登录源IP地址。

**相关命令**

**/usr/bin/last root 2>/dev/null | /bin/awk '$2 !~ /^(console|tty1|:0.\*)$/ && $3 !~ /^(console|tty1|:0.\*)$/{print}'**

## 2.18 监控审计服务器正在监听的端口

**相关命令：**

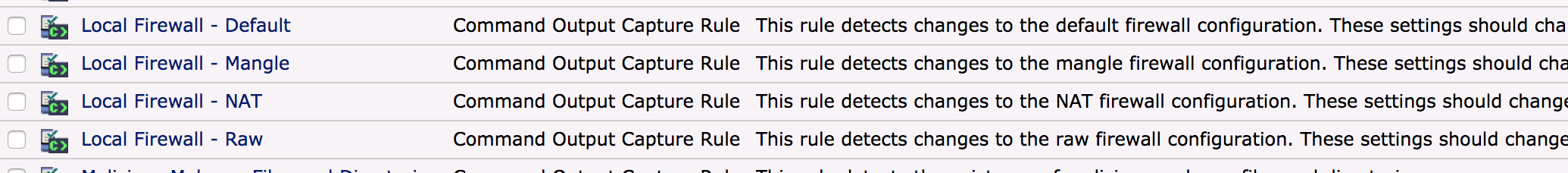
**/bin/netstat --inet -nl 2>/dev/null | /bin/awk '$1 !~ /^(Proto|Active)/{print "Protocol: " $1 "\t\t\tForeign Address: " $5 "\t\tLocal Address: " $4}' | /bin/sort**

## 2.19 监控 nologin shell的用户，shell不被更改

**相关命令**

**/bin/awk -F: '$1 !~ /^[[:space:]]\*#/ {print $1 ":" $3 ":" $4 ":" $6 ":" $7}' /etc/passwd 2>/dev/null**

## 2.20 监控防火墙配置变更

****

## 2.21 监控是否存在恶意程序

**相关命令（持续维护木马名称更新）:**

**MaliciousFileEntries="55808 Trojan - Variant A:/tmp/.../a,/tmp/.../,/tmp/.../r;ADM W0rm:/tmp/.w0rm;Adore:/usr/lib/lib,/etc/cron.daily/0anacron;BeastKit:/usr/include/rpc/ ../,/usr/include/rpc/ ../kit,/usr/include/rpc/ ../kit2,/usr/doc/.sl,/usr/doc/.sp,/usr/doc/.statnet,/usr/doc/.logdsys,/usr/doc/.dpct,/usr/doc/.gifnocfi,/usr/doc/.dnif,/usr/doc/.nigol,/lib/ldd.so/bktools/;BOBKit:/usr/sbin/ntpsx,/usr/sbin/.../,/usr/lib/.../,/usr/include/.../;CiNIK Worm (Slapper.B variant):/tmp/.cinik;Dica:/var/run/...dica/,/lib/.sso,/lib/.so;illogic:/lib/secuity/.config/;Jynx:/xochikit/,/omgxochi/;Kbeast:/usr/\_h4x\_/;Knark:/proc/knark/;Phalanx2:/etc/khubd.p2/,/etc/lolzz.p2/,/usr/lib/zupzz.p2/;Ramen Worm:/usr/src/.poop/,/sbin/asp;Scalper Worm:/tmp/.uua,/tmp/.a;Slapper:/tmp/.bugtraq,/tmp/.bugtraq.c,/tmp/.unlock,/tmp/httpd,/tmp/update,/tmp/.cinik,/tmp/.b;T0rn Rootkit:/usr/.lib/,/usr/src/.puta,/dev/.lib/,/usr/info/.t0rn;zaRwT.KiT Rootkit:/dev/ttyn,/dev/ttyp,/dev/ttyf,/dev/ttyl"; SavedIFS="$IFS"; IFS=";"; for FileEntry in $MaliciousFileEntries; do MalwareName=`/bin/echo "$FileEntry" | /bin/awk -F: '{print $1}'`; FileList=`/bin/echo "$FileEntry" | /bin/awk -F: '{print $2}'`; IFS=","; IsExisted=`/bin/ls -d $FileList 2>/dev/null`; if [ -n "$IsExisted" ]; then /bin/echo -e "$MalwareName:\n$IsExisted"; fi; IFS=";"; done; IFS="$SavedIFS";**

## 2.22 监控是否存在恶意程序端口

**PortDatabase="65535-#Adore;6660-TCP:6661-TCP:6662-TCP:6663-TCP:6664-TCP:6665-TCP:6666-TCP:6667-TCP:6668-TCP:6669-TCP:6697-TCP:7000-TCP:7001-TCP#Generic IRC Botnet;14856-TCP#TuxKit;52901-UDP#Omega DDoS Tool;27374-#Ramen Worm;2001-UDP#Scalper Worm;2002-TCP:4156-TCP:1978-TCP:1812-TCP:2015-TCP#Slapper;32982-TCP#Solaris Wanuk;2555-TCP:47017-TCP#T0rn Rootkit;60922-TCP#zaRwT.KiT Rootkit"; TCP\_Ports=`/bin/netstat -an 2>/dev/null | /bin/awk '/^tcp[[:space:]]+.\*/,/^$/ { print $4 }'`; UDP\_Ports=`/bin/netstat -an 2>/dev/null | /bin/awk '/^udp[[:space:]]+.\*/,/^$/ {print $4}'`; SavedIFS="$IFS"; IFS=";"; for PortEntry in $PortDatabase; do MalwareName=`/bin/echo "$PortEntry" | /bin/awk -F# '{print $2}'`; MalwarePortInfo=`/bin/echo "$PortEntry" | /bin/awk -F# '{print $1}'`; IFS=":"; OpenPort=""; for PortInfo in $MalwarePortInfo; do Port=`/bin/echo "$PortInfo" | /bin/awk -F- '{print $1}'`; Protocol=`/bin/echo "$PortInfo" | /bin/awk -F- '{print $2}'`; TCPOpened=`/bin/echo "$TCP\_Ports" | /bin/egrep ":$Port$"`; UDPOpened=`/bin/echo "$UDP\_Ports" | /bin/egrep ":$Port$"`; if [ "$Protocol" = "TCP" -a -n "$TCPOpened" ]; then OpenPort="$OpenPort $TCPOpened - TCP"; fi; if [ "$Protocol" = "UDP" -a -n "$UDPOpened" ]; then OpenPort="$OpenPort $UDPOpened - UDP"; fi; if [ "$Protocol" = "" ]; then if [ -n "$TCPOpened" ]; then OpenPort="$OpenPort $TCPOpened - TCP"; fi; if [ -n "$UDPOpened" ]; then OpenPort="$OpenPort $UDPOpened - UDP"; fi; fi; done; IFS=";"; if [ -n "$OpenPort" ]; then OpenPort=`/bin/echo "$OpenPort" | /bin/awk '{Content = Content " " $0;}END{print Content}'`; /bin/echo "Malware Name: $MalwareName"; /bin/echo "Open Port(s): $OpenPort"; fi; done; IFS=$SavedIFS;**

## 2.23 监控由未知用户启动的进程

**相关命令：**

**if [ -e "/usr/sbin/lsof" ]; then Cmd="/usr/sbin/lsof"; else if [ -e "/usr/bin/lsof" ]; then Cmd="/usr/bin/lsof"; fi; fi; if [ -n "$Cmd" ]; then UserIDs=`(/bin/cat /etc/passwd 2>/dev/null; /usr/bin/ypcat passwd 2>/dev/null) | /bin/awk -F: -v ORS=';' '$0 !~ /^[[:space:]]\*#/{print $3}'`; $Cmd -n -P -l 2>/dev/null | /bin/awk 'BEGIN{UIDs = "'$UserIDs'"}{Regex="(^|;)" $3 "(;|$)" ; if(UIDs !~ Regex){print} }'; else /bin/echo "[lsof] utility is not found."; fi;**

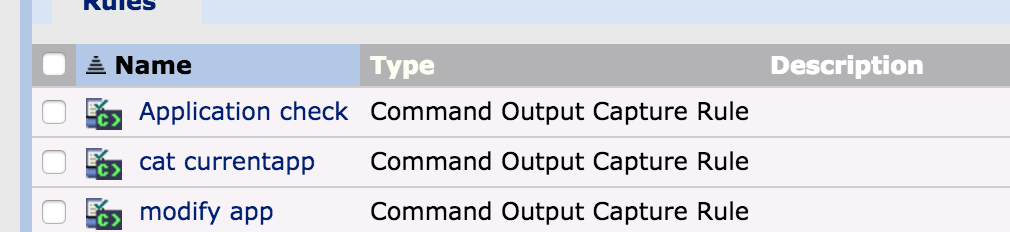
## 2.24 检测重复组ID与UID

****

****

**可通过伪造重复组ID 进行越权**

## 2.25 应用程序白名单监控，对不信任的进程名进行告警。

****

## 2.26 监控nginx/apache 目录配置文件改动