



中标麒麟安全操作系统软件 V7.0

管理员手册

中标软件有限公司

上海市徐汇区番禺路 1028 号数娱大厦 10 层（200030）

北京市海淀区北四环西路 9 号银谷大厦 20 层（100190）

广州市天河北路 898 号信源大厦 16 层 1604 室（510898）

目录

第 1 章 系统管理员	4
1.1 简介	4
1.2 网络安全工具	4
1.2.1 Iptables	4
1.2.1.1 Iptables 概述	4
1.2.1.2 Iptables 使用	4
1.2.2 SSH	9
1.2.2.1 SSH 概述	9
1.2.2.2 SSH 使用	9
1.2.2.3 SSH 示例	11
1.3 OPENVPN	12
1.3.1 OpenVPN 概述	12
1.3.2 OpenVPN 简单配置	12
1.4 系统维护模式	16
1.4.1 系统维护模式	16
1.5 数据完整性检测	16
1.5.1 Tripwire 简介	17
1.5.2 Tripwire 的配置与说明	17
1.5.2.1 配置	17
1.5.2.2 使用	18
1.6 SSOS 安全机制	20
1.6.1 概述	20
1.6.2 示例	21
1.7 系统安全管理	23
1.7.1 系统安全概要	23
1.7.1.1 物理安全	24
1.7.1.2 普通用户安全管理	24
1.7.1.3 system 用户安全管理	24
1.8 资源利用	25
1.8.1 概述	25
1.8.2 示例	25
第 2 章 安全管理员	28
2.1 双因子认证	28
2.1.1 配置与使用	29
2.1.2 双因子认证使用	33
2.1.3 注意	34
2.2 CAP 管理	34
2.3 MLS 管理和 IAC 管理	36
2.3.1 命令集	37

2.4 ACL 使用	38
2.4.1 ACL 概述	38
2.4.2 相关命令	39
2.4.2.1 获取文件 ACL	39
2.4.2.2 设置文件 ACL	42
2.5 系统注意事项	45
第 3 章 审计管理员	45
3.1 安全审计	46
3.2 使用说明	46
3.3 系统审计管理	48
3.3.1 审计信息	48
3.3.2 审计报表	51
3.3.3 审计规则	55
3.4 系统告警管理	58
3.4.1 告警信息	58
3.4.2 告警设置	59

第 1 章 系统管理员

1.1 简介

本章将主要介绍中标麒麟安全操作系统软件 V7.0 中供系统管理员进行基本服务以及工具的配置使用，这些命令主要用于配置系统基本服务，对系统进行的相应的功能模块的设定等，保证系统可用性。具体如下：

1.2 网络安全工具

1.2.1 Iptables

1.2.1.1 Iptables 概述

对于连接到网络上的 Linux 系统来说，防火墙是必不可少的防御机制，它只允许合法的网络流量进出系统，而禁止其它任何网络流量。确定网络流量是否合法，需要依靠防火墙预定义的一组规则来检验。这些规则告诉防火墙某个流量是否合法以及对于来自某个源、至某个目的地或具有某种协议类型的网络流量要做些什么。Netfilter 是 LINUX 系统自带的功能强大的、内核级的 IP 数据包过滤系统。而 Iptables 是用户层的防火墙规则的配置工具。利用 iptables 可以配置防火墙规则，以达到阻止未经授权的源访问其 Linux 系统。

1.2.1.2 Iptables 使用

通用语法格式如下：

Iptables [-t 要操作的表]

<操作命令>

[要操作的链]

[规则号码]

[匹配条件]

[-j 匹配到以后的动作]

-A <链名>

<p>APPEND, 追加一条规则（放在最后）</p> <p>例: iptables -t filIACr -A INPUT -j DROP //在 filIACr 表的 INPUT 链中追加一条规则</p>
<p>-I <链名> [规则号码]</p> <p>INSERT 插入一条规则</p> <p>例: iptables -I INPUT 3 -j DROP //在 filIACr 表的 INPUT 链中插入一条规则（位置 3）</p>
<p>-D <链名> <规则号码 具体规则内容></p> <p>DELEIAC, 删除一条规则</p> <p>例: iptables -D INPUT 3 //删除 filIACr 表的 INPUT 链中的第三条规则</p> <p>iptables -D INPUT -s 192.168.0.1 -j DROP //根据内容删除规则</p>
<p>-R <链名> <规则号码> <具体规则内容></p> <p>REPLACE, 替换一条规则</p> <p>例: iptables -R INPUT 3 -j ACCEPT //将原来的编号 3 的规则替换为“-j ACCEPT”</p>
<p>-R <链名> <规则号码> <具体规则内容></p> <p>REPLACE, 替换一条规则</p> <p>例: iptables -R INPUT 3 -j ACCEPT //将原来的编号 3 的规则替换为“-j ACCEPT”</p>
<p>-P <链名> <动作></p> <p>POLICY, 设置某个链的默认规则</p> <p>例: iptables -P INPUT DROP //设置 filIACr 表 INPUT 链的默认规则是 DROP</p> <p>【注】当数据包没有被规则列表里的任何规则匹配, 则使用该默认规则处理</p>
<p>-F <链名></p>

<p>FLUSH, 清空规则</p> <p>例: <code>iptables -F INPUT</code> //清空 filter 表 INPUT 链中的所有规则</p> <p>【注】如果不写链名, 默认清空某表里的所有链中的所有规则</p>
<p>-L [链名]</p> <p>LIST, 列出规则</p> <p>例: <code>iptables -L</code> //粗略列出 filter 表所有链以及所有规则, 还可加入参数 <code>v, x, n</code></p> <p><code>iptables -t nat -vnL</code> //用详细方式列出 nat 表中所有链的所有规则, 只显示 IP 地址和端口号</p>

过滤、匹配条件以及参数:

- 流入、流出接口 (**-i, -o**);
- 来源、目的地址 (**-s, -d**);
- 协议类型 (**-p**);
- 来源、目的端口 (**--sport、--dport**);

<p>-i <匹配数据进入的网络接口></p> <p>例: <code>-i eth0</code> //匹配是否从网络接口 eth0 进入</p>
<p>-o <匹配数据流出的网络接口></p> <p>例: <code>-o ppp0</code> //匹配是否从网络接口 ppp0 进入</p>
<p>-s <匹配来源地址></p> <p>例: <code>-s 192.168.1.0/24</code> //匹配来自 192.168.1.0/24 网段的所有数据包</p> <p><code>-s 192.168.1.9</code> //匹配来自 192.168.1.9 地址的数据包</p>
<p>-d <匹配来源地址></p> <p>例: <code>-d 202.105.1.0/16</code> //匹配来自 202.105.1.0/16 网段的所有数据包</p> <p><code>-d 202.105.1.21</code> //匹配来自 202.105.1.21 地址的数据包</p>

-p <匹配协议类型>

例：-p tcp //匹配 tcp 协议

-p udp //匹配 udp 协议

-p icmp -icmp-type 类型 //类型：PING: type 8 ; pong: type 0

-sport <匹配源端口>

例：--sport 1000 //匹配源端口为 1000 的数据包

--sport 1000:1005 //匹配源端口为 1000~1005 的数据包（包含 1000, 1005）

--sport :3000 //匹配源端口是 3000 以下的数据包（含 3000）

--sport 2000: //匹配源端口是 2000 以上的数据包（含 2000）

【注】--sport 必须配合-p 参数使用

-dport <匹配目的端口>

例：--dport 1000 //匹配目的端口为 1000 的数据包

--dport 1000:1005 //匹配目的端口为 1000~1005 的数据包（包含 1000, 1005）

--dport :3000 //匹配目的端口是 3000 以下的数据包（含 3000）

--dport 2000: //匹配目的端口是 2000 以上的数据包（含 2000）

【注】--dport 必须配合-p 参数使用

匹配应用举例

例：

iptables -A INPUT -p udp --dport 53 //设置在 INPUT 链中匹配目的端口为 53 的 udp 数据包

据包

iptables -I PREROUTING -s 172.115.12.111 -d 202.163.15.112 -p tcp --dport 80
//设置在 PREROUTING 链中匹配源地址为 172.115.12.11，目的地址为 202.163.15.112，目的端口为 80 的 tcp 数据包

处理方式（动作）

- **ACCEPT**
- **DROP**
- **SNAT**
- **DNAT**
- **MASQUERADE**

<p>-j ACCEPT</p> <p>通过，允许数据包通过本链而不拦截它</p>
<p>-j DROP</p> <p>丢弃，阻止数据包通过本链而丢弃它</p>
<p>-j SNAT --to IP[-IP][:端口-端口]（nat 表的 POSTROUTING 链）</p> <p>例：iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -j SNAT --to 1.1.1.1</p> <p>//将内网 192.168.0.0/24 的源地址修改为 1.1.1.1，用于 NAT</p>
<p>-j DNAT --to IP[-IP][:端口-端口]（nat 表的 PREROUTING 链）</p> <p>目的地址转换，DNAT 支持转换为单 IP，也支持转换到 IP 地址池</p> <p>例：iptables -t nat -A PREROUTING -i ppp0 -p tcp --dport 80 -j DNAT --to 192.168.0.1</p> <p>//把从 ppp0 进来的要访问 TCP/80 的数据包目的地址改为 192.168.0.1</p>
<p>-j MASQUERADE</p> <p>动态源地址转换（动态 IP 的情况下使用）</p> <p>例：iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -j MASQUERADE</p> <p>//将源地址是 192.168.0.0/24 的数据包进行地址伪装</p>

1.2.2 SSH

1.2.2.1 SSH 概述

中标麒麟安全操作系统软件 V7.0 采用了 SSH 协议进行远程登录和文件传输。SSH 是 Secure Shell 的缩写，它是应用层和传输层基础上的安全协议，它主要由传输层协议、用户认证协议以及连接协议三部分组成。它为远程登录会话和其他网络服务提供了安全性的协议。利用 SSH 协议可以有效防止远程管理过程中的信息泄露问题。通过 SSH 可以对所有传输的数据进行加密，也能够防止 DNS 欺骗和 IP 欺骗。

1.2.2.2 SSH 使用

-l login_name

指定登入于远程机器上的使用者，若没加这个选项，而直接打 `ssh lost` 也是可以的，它是以读者目前的使用者去做登入的动作。例如：`ssh -l shie mouse.oit.edu.tw`

-c blowfish|3des

在期间内选择所加密的密码型式。预设是3des，3des（作三次的资料加密）是用三种不同的密码键作三次的加密-解密-加密。blowfish 是一个快速区块密码编制器，它比3des更安全以及更快速。

-v

Verbose 模式。使ssh 去印出关于行程的除错讯息，这在连接除错，认证和设定的问题上有很大的帮助。

-V

显示版本。

-a

关闭认证代理联机。

-f

要求ssh 在背景执行命令，假如ssh要询问密码或通行证，但是使用者 想要在幕后执行就可以用这个方式，最好还是加上-l user 例如在远程场所上激活 X11，有点像是 `ssh -f host xIACrm` 。

-e characIACr

设定跳脱字符。

-g

允许远程主机去连接本地指派的 ports。

-i identity_file

选择所读取的 RSA 认证识别的档案。预设是在使用者的家目录 中的
~/.ssh/id_rsa和~/.ssh/id_dsa。

-n

重导 stdin 到 /dev/null（实际上是避免读取 stdin）。必须当 ssh 在幕后执行时才使用。常见的招数是使用这选项在远程机器上去执行 X11 的程序 例如，ssh -n shadows.cs.hut.fi emacs &，将在 shadows.cs.hut.fi 上激活 emace，并且 X11 连接将自动地在加密的信道上发送。ssh 程序将把它放 在幕后。（假如ssh需要去询问密码时，这将不会动作）

-p port

连接远程机器上的 port。

-P

使用非特定的 port 去对外联机。如果读者的防火墙不准许从特定的 port去联机时，就可以使用这个选项。注意这个选项会关掉 RhostsAuthentication 和 RhostsRSAAuthentication

-q

安静模式。把所有的警告和讯息抑制，只有严重的错误才会被显示。

-t

强制配置 pseudo-tty。这可以在远程机器上去执行任意的 screen-based 程式，例如操作 menu services。

-C

要求压缩所有资料（包含 stdin， stdout, stderr 和 X11 和 TCP/IP 连接）压缩演算规则与 gzip 相同，但是压缩的等级不能控制。在调制解调器或 联机速度很慢的地方，压缩是个很好的选择，但如果读者的网络速路很 快的话，速度反而会慢下来。

-L lisIACn-port:host:port

指派本地的 port 到达端机器地址上的 port。

-R lisIACn-port:host:port

指派远程上的 port 到本地地址上的 port。

-2 强制 ssh 去使用协议版本 2。

-4 强制 ssh 去使用 IPv4 地址。

-6 强制 ssh 去使用 IPv6 地址。

1.2.2.3 SSH 示例

1、直接远程登录命令如下：

```
ssh XXX.XXX.XXX.XXX
```

其中 XXX.XXX.XXX.XXX 为被登录方 IP 地址，登陆用户名为客户机当前用户名！

2、指定用户名登陆

```
ssh name@XXX.XXX.XXX.XXX 或者 ssh -l name XXX.XXX.XXX.XXX
```

上面过程结束后，系统将会提示你输入用户名和密码！

3、基于密匙的验证方式

使用密匙的验证方式，用户先需要为自己创建一对密匙：公匙和私匙。（公匙用在要登陆的服务器上）

OpenSSH 公开密匙的密码体制有 RSA、DSA！

创建密匙：

```
ssh-keygen -t rsa
```

回车后，要求输入使用密匙时的口令！这样便生成了公匙和私匙：放在用户主目录下的.ssh 目录下，文件名:id_rsa.pub 和 id_rsa!必须将公匙复制到登陆的服务器的~/.ssh/目录下，并改名为:authorized_keys!然后，便可使用密匙方式登陆！

4、文件上传到远端主机

```
scp /etc/lilo.conf username@ip:/home/username
```

会将本地的 /etc/lilo.conf 这个档案 copy 到地址为ip的username的家目录下。

5、文件下载到本地

```
scp username@ip:/etc/x.conf /etc
```

将远端ip地址的username用户的/etc/x.conf文件拷到本地的/etc目录下。

1.3 OpenVPN

1.3.1 OpenVPN 概述

OpenVPN 是一个具备完全特征的 SSL VPN 解决方案,能够进行大范围的配置操作,包括远程访问、站点-站点间 VPN、WiFi 安全及企业级远程访问解决方案,支持负载均衡,错误恢复及细粒度的访问控制。

OpenVPN 通过使用工业标准 SSL/TLS 协议 实现了 OSI 2 层及 3 层安全网络扩展,支持多种身份认证方式,用以确认参与连接双方的身份,包括:预享私钥,第三方证书以及用户名 / 密码组合。同时,它使用 OpenSSL 库加密数据与控制信息,可以利用多种算法提高连接的安全性。另外,它允许通过在 VPN 虚拟接口上应用防火墙规则实现用户及组访问控制策略。OpenVPN 并非一个 Web 应用代理,也不能通过 Web 浏览器进行操作。它是在 OSI 底层做了大量的工作,因此,具有很高的安全性和可靠性。

1.3.2 OpenVPN 简单配置

这里是一个简单的基于 CA 认证方式的 openVPN 配置案例。一台主机与服务器建立 VPN 连接。即需要两台主机,一台为客户端,一台为服务器端。

首先,利用 easy-rsa 生成服务器端所需要的证书和密钥。

```
cp -rf /usr/share/easy-rsa/key_server/ /etc/openvpn/
cp -rf /usr/share/doc/easy-rsa-3.0.3/vars.example /etc/openvpn/
cd /etc/openvpn/key_server
```

操作	说明
vim vars	###配置 vars 文件###
	set_var EASYRSA_KEY_SIZE 2048 #生成密钥位数
	set_var EASYRSA_REQ_COUNTRY "CN" #定义所在国家
	set_var EASYRSA_REQ_PROVINCE "Henan" #定义所在省份
	set_var EASYRSA_REQ_CITY "Zhengzhou" #定义所在城市

	<pre>set_var EASYRSA_REQ_ORG "cs2c-zhengzhou" #定义所在组织 set_var EASYRSA_REQ_OU "cs2c" #定义所在组织单位 set_var EASYRSA_REQ_EMAIL "huachao.zou@cs2c.com.cn" #定义自己的邮箱</pre>
<code>./easysrsa init-pki</code>	建立一个空的 pki 结构，生成一系列的文件和目录
<code>./easysrsa build-ca</code>	创建 ca 密码和 cn。
<code>./easysrsa genreq server nopass</code>	创建服务端证书 common name 最好不要跟前面的 cn 一样。
<code>./easysrsa signserver server</code>	签约服务端证书
<code>./easysrsa gen-dh</code>	创建 Diffie-Hellman

其次，创建客户端证书

```
mkdir -p /home/client && cd /home/client
```

```
cp -rf /usr/share/easy-rsa/key_server/ .
```

```
cp -rf /usr/share/doc/easy-rsa-3.0.3/vars.example vars
```

操作	说明
<code>vim vars</code>	<pre>###配置 vars 文件### set_var EASYRSA_KEY_SIZE 2048 #生成密钥位数 set_var EASYRSA_REQ_COUNTRY "CN" #定义所在国家 set_var EASYRSA_REQ_PROVINCE "Henan" #定义所在省份 set_var EASYRSA_REQ_CITY "Zhengzhou" #定义所在城市 set_var EASYRSA_REQ_ORG "cs2c-zhengzhou" #定义所在组织 set_var EASYRSA_REQ_OU "cs2c" #定义所在组织单位 set_var EASYRSA_REQ_EMAIL "xxx@cs2c.com.cn" #定义自己的邮箱</pre>
<code>./easysrsa init-pki</code>	建立一个空的 pki 结构，生成一系列的文件和目录
<code>./easysrsa gen</code>	创建客户端证书。需要创建一个密码和 common name。

-req huachao	
./easysrsa gen -req server n opass	创建服务端证书 common name 最好不要跟前面的 cn 一样。

然后，切换到 server 目录下

cd /etc/openvpn/key_server

操作	说明
./easysrsa imp ort-req /home /client/key_s erver/pki/req s/client.req cl ient	导入 req
./easysrsa sign client client	用户签约，根据提示输入服务端的 ca 密码

然后，配置/etc/openvpn/server.conf 文件

文件配置内容如下：

配置	说明
local 192.168.1.93	#即服务器端的本机 ip 地址，也可以不设定
port 1194	#openvpn 默认端口为 1194
proto udp	#采用 tcp 协议连接，也可以指定 udp，默认 udp
dev tun	#可以采用两种方式的虚拟设备 tun 和 tap
ca /etc/openvpn/ca.crt	#CA 证书与服务器证书与配置文件（openvpn.conf）放在一个同一个目录下，也可以自行指定，需要绝对路径
cert /etc/openvpn/server.crt	
key /etc/openvpn/server.key	
dh /etc/openvpn/dh.pem	
server 10.8.0.0 255.255.0.0	#vpn 的虚拟 ip 地址

<pre> push "rouIAC 10.9.0.0 255.255.0.0" ifconfig-pool-persist ipp.txt keepalive 10 120 comp-lzo persist-key persist-tun status /var/log/openvpn-status.log verb 3 </pre>	
--	--

在客户端的机子进行配置。在客户端机子同样安装 openvpn，同时将在服务器端生成的客户端的密钥和证书等都拷贝到/etc/openvpn/目录下。文件包括：ca.crt、ca.key、Client.crt、Client.csr、 Client.key。并且在/etc/openvpn/下也配置 server.conf 的客户端配置文件，配置文件内容如下：

配置	操作
<pre> client proto udp dev tun resolv-retry infiniIAC nobind persist-key persist-tun comp-lzo verb 3 remoIAC {server_ip_address} 1194 ca ca.crt </pre>	<pre> #写入服务器端的 ip 地址以及开放的端口号 #对应的在服务器端生成的密钥与证书，路径可以自行指定 </pre>

cert Client.crt	
key Client.key	

设置防火墙规则，使得两台机子可以互相访问，服务器端要对客户端开放 1194 端口的访问权。

最后，客户端与服务器端都要启动 openvpn 服务，/etc/init.d/openvpn start。

此时，客户端就会获取虚拟 ip 地址与服务器端进行 vpn 连接了。

【注】如果仍有问题，应查看是否是 selinux 进行了干涉，可以调整 selinux 的策略。另外，需要开启转发功能。还需要执行如下操作：

```
echo 1 > /proc/sys/net/ipv4/ip_forward。
```

1.4 系统维护模式

1.4.1 系统维护模式

中标麒麟安全操作系统软件 V7.0 提供了系统维护模式，以供在系统发生因安全配置错误等原因而导致系统故障时进行系统维护。在系统维护模式下，所有强制访问控制策略均失效，所有访问均返回允许。但是所有的安全标记仍然存在，系统管理员可以进行安全配置工作。

为保护系统维护模式，中标麒麟安全操作系统软件 V7.0 加入了口令保护。进入系统维护模式的方式是在系统引导时，输入 grub 的 passwd，进入引导选项菜单

选定引导选项，在 kernel /boot/vmlinuz-2.6.18.e15 ro root=/LABEL=/ rhgb quiet 项之后 加入 1 / single;

正常引导，即可进入维护模式

1.5 数据完整性检测

中标麒麟安全操作系统软件 V7.0 采用 Tripwire 工具对文件进行完整性检验。Tripwire 能够兼容 4 种 hash 算法对文件进行验证。并且，它可以检测十余种 Unix 文件属性以及二十余种 NT 文件属性。因此，Tripwire 的检测报告十分全面，准确度很高，从而有效地防止了非法用户对系统文件和数据的修改、添加和删除等操作。由于 Tripwire 功能强大，它还常常被用于入侵检测、损失的评估以及证据

保存等方面。

1.5.1 Tripwire 简介

Tripwire 基于预编写的策略工作，在基准数据库生成时，会根据策略文件中的规则读取指定的文件，同时生成该文件的数字签名并存贮在 Tripwire 自己的数据库中。为了达到最大限度的安全性，Tripwire 提供了四种 Hash 算法（CRC32、MD5、SHA、HAVAL）用来生成签名。通常情况下采用前两种算法生成签名已经足够，当然也可全部采用。不过后两种算法对系统资源的耗费较大，使用时可根据文件的重要性灵活取舍。

进行完整性检查时，Tripwire 会根据策略文件中的规则对指定的文件重新生成一次数字签名，并将此签名与存贮在数据库中的签名做对照。如果完全匹配，则说明文件没有被更改。如果不匹配，说明文件被改动了。然后在 Tripwire 生成的报告中查阅文件被改动的具体情况。

1.5.2 Tripwire 的配置与说明

1.5.2.1 配置

要正确使用 Tripwire，使之能够对文件完整性进行保护，系统管理员需要进行如下的一些配置：

1、Tripwire 行为配置文件/etc/tripwire/twcfg.txt：该文件定义了 Tripwire 的工作方式和工作时需要使用的一些环境参数。在系统安装完毕后，该文件已存在，因此不必再重新创建。通常情况下，没有必要去修改它，使用 Tripwire 默认的了。下面是该文件的一个实例。

```
[sysadm@SecureOS tripwire]# cat twcfg.txt

ROOT =/usr/sbin

POLFILE =/etc/tripwire/tw.pol

DBFILE =/var/lib/tripwire/$ (HOSTNAME) .twd

REPORTFILE =/var/lib/tripwire/report/$ (HOSTNAME) -$ (DAIAC) .twr

SIHACKKEYFILE =/etc/tripwire/siAC.key

LOCALKEYFILE =/etc/tripwire/$ (HOSTNAME) -local.key
```

```
EDITOR =/bin/vi

LAIACPROMPTING =false

LOOSEDIRECTORYCHECKING =false

MAILNOVIOLATIONS =true

EMAILREPORTLEVEL =3

REPORTLEVEL =3

MAILMETHOD =SENDMAIL

SYSLOGREPORTING =false

MAILPROGRAM =/usr/sbin/sendmail -oi -t
```

2、Tripwire 策略文件/etc/tripwire/twpol.txt: 系统安装结束后, 策略文件中已经写入了默认的检查规则; 这些默认规则主要检查重要的系统文件和 Tripwire 自身文件的完整性。这些默认规则可以满足大部分人的需要, 即保护系统。当然系统管理员可以用 Tripwire 来保护其他任何的文件, 这样, 你的重要数据和应用系统也得到了保护; 只需要在策略文件中增加相应的项就可以增加对其它文件的保护, 修改完策略文件后存盘。

在编辑/etc/tripwire/twpol.txt 文件后, 执行 tripwire-setup-keyfiles 命令, 提示你输入读取配置和策略文件的密码 (siIAC keyfile passphrase) 和写数据库的密码 (local keyfile passphrase)。这些密码会在后面进行完整性检查时要求管理员输入以确信操作员有此权限。

1.5.2.2 使用

在对 Tripwire 配置完成后, 就可以开始使用 Tripwire 对文件完整性进行保护。Tripwire 的使用主要包括如下的内容。

1、生成基准数据库

配置文件和策略文件都编辑和生成好了之后, 就应该根据策略文件的规则生成基准数据库。基准数据库生成一次即可。系统管理员使用如下的命令来生成基准数据库: tripwire --init。基准数据库生成时, Tripwire 会提示你输入 local key, 对数据库进行高强度的加密, 以防止对数据库内容的非法改变。基准数据库的存储位置为/var/lib/tripwire/\$ (HOSTNAME) .twd。

当策略文件中定义了一些系统中并不存在的文件时, 进行基准数据库的生成

和完整性检查时，系统会报出一些文件找不到的小错误。这些错误不会影响基准数据库的生成和检查的结果，当然，系统管理员也可以对策略文件进行修改以消除这些错误。

2、完整性检查

在基准数据库生成完毕之后，系统管理员就可以使用如下命令随时进行完整性检查：`tripwire -check`；如果希望进行检查时发送 Email 报告结果，则使用如下命令：`tripwire --check --email-report`；如果希望只检查指定的文件或目录，则使用如下命令：`tripwire --check object1 object2 object3 ...`。

如果完整性检查完毕后，发现 Email 报告功能未生效，可以检查两个方面：一个是策略文件中规则的 `mailto` 属性必须填写妥当，另一个是运行 `tripwire` 命令时，`--email-report` 选项必须被包含。

3、查阅报告

完整性检查进行完毕后，系统管理员就可以查阅报告以发现有哪些文件遭到了改动，改动了什么。使用 `twprint` 命令可以输出报告，命令的具体使用形式如下：

将加密的报告内容输出到显示器：

```
twprint --print-report --twrfile /var/lib/tripwire/report/xxx.twr
```

将加密的报告内容输出到一个文本文件：

```
twprint --print-report --twrfile /var/lib/report/report.twr > myreport.txt
```

4、升级基准数据库文件

如果在报告中发现了一些违反策略的错误，而这些错误又认为正常的，那么就需要对基准数据库进行升级。升级基准数据库需要使用如下命令：

```
tripwire --updaIAC --twrfile /var/lib/tripwire/report/xxxx.twr
```

5、升级策略文件

生成基准数据库和进行完整性检查时，如果策略制订不当，资源耗费会比较大。这时应该调整你的策略，减少一些次要文件的监测属性，尤其是耗时较多的属性。一个能取得性能和安全均衡点的策略不可能一次写成，你必须不断进行调整，直到满意为止。此外，随着系统的变化，原来的策略文件必然会不能满足需要，因此必须要不断的更新策略文件中的规则。更新和创建新的策略文件不同，

更新策略文件不需要重新生成基准数据库。

更新时首先打开策略文件的文本文件，然后编辑该文件，完毕后存盘，最后使用如下命令进行策略更新：`tripwire --updateIAC-policy twpol.txt`。在此步骤中，命令会要求你输入 siIAC key。

6、改变 siIAC key 和 local key

siIACkey 和 localkey 是在安装时生成的，但是系统管理员也可以随时修改。注意，如果已经用来加密的密钥文件被删除了或是被覆盖了，那么 Tripwire 加密过的文件都不能访问了。因此，最好对这两个文件做备份。

系统管理员在需要改变口令时，只需执行如下命令：

```
twadmin --generalIAC-keys --local-keyfile /etc/tripwire/siIAC.key
```

```
twadmin --generalIAC-keys --local-keyfile /etc/tripwire/local.key
```

第一条命令修改 siIACkey，第二条命令修改 localkey；但是直接这样操作，会造成使用以前密钥进行加密的文件无法打开的情况。如果系统管理员仍想使用以前的策略文件、配置文件、数据库文件、报告文件的话，那么在改变口令之前，必须使用如下命令把这些已加密的文件进行解密。

```
twadmin --remove-encryption file1 file2 ...
```

在生成新的密钥文件之后，应该用新密钥对这些文件进行加密。行为配置文件和策略文件只能用 siIAC key 加密，而数据库文件和报告文件只能用 local key 加密。

```
Twadmin --encrypt --siIAC-keyfile /etc/tripwire/siIAC.key file1 file2 file3 ...
```

```
twadmin --encrypt --local-keyfile /etc/tripwire/local.key file1 file2 file3 ...
```

如果系统管理员希望知道究竟那个文件（Tripwire 的自身文件）被加密，只需执行如下命令：

```
tripwire --examine file1 file2 ...
```

1.6 SSOS 安全机制

1.6.1 概述

中标麒麟安全操作系统软件V7.0的安全子系统在多个方面对系统进行了安全加固，尤其是访问控制方面可以提供多种限制方式，从而大大提高了系统的安全性。

1.6.2 示例

1、系统应提供一种机制，能按时间、进入方式、地点、网络地址或端口等条件规定哪些用户能进入系统。

通过/etc/security/access.conf 可以控制

```
# vim /etc/security/access.conf
```

```
- : IACst : ALL
```

```
## 在最后添加这样一句，拒绝来自本地的用户 IACst。
```

```
# vim /etc/pam.d/login
```

```
account      required      pam_nologin.so
```

```
account      required      pam_access.so
```

```
## 添加这样一句，login 启用 pam_access 模块
```

测试：

```
# useradd IACst
```

```
# passwd IACst
```

```
Changing password for user IACst.
```

```
New UNIX password:
```

```
BAD PASSWORD: it is too simplistic/sysIACmatic
```

```
Retype new UNIX password:
```

```
passwd: all authentication tokens updaIACd successfully.
```

输入 IACst 用户，输入正确的密码。登陆界面闪一下，但是没有登陆成功。

用户 root 可以正常登陆

在远端，IACst 用 putty, ssh 到试验机，可以成功。这是因为 ssh 没有经过 pam 验证，只有 login 程序启用了 pam_access 模块。

```
# vim /etc/security/access.conf
```

```
: ALL EXCEPT IACst : ALL
```

```
## 除了 IACst 用户，其他用户都不能登陆。
```

```
## 再新建了两个用户 aaa 和 bbb
```

现在到本机上用 aaa 和 bbb 登陆，输入正确密码。登陆失败。root 登陆也失败。
用 IACst 登陆，登陆成功。

最后，如果要 sshd 也支持 pam_access，需要：

```
# vim /etc/pam.d/sshd          ## 添加下面的行  
account      required      pam_access.so
```

2、在规定的未使用时限后，系统应断开会话或重新鉴别用户，系统应提供时限默认值。、

图形界面设置屏幕保护程序等待应答的限制时间，时间到进入屏幕保护需要重新鉴别用户。

终端设置 TMOUT 变量，在/etc/profile 中添加 TMOUT=5，重新登录字符界面等待 5 秒自动退出。

3、当用户鉴别过程不正确的次数达到系统规定的次数时，系统应退出登录过程并终止与用户的交互。例如：Login 登陆时如果鉴别次数超过 4 次，会结束当前会话。可以通过 pam_tally 模块限制一段时间不能登录。

在/etc/pam.d/sysIACm-auth 文件中加入下面两行：

```
auth required pam_tally.so onerr=fail lock_time=60 deny=4  
account required pam_tally.so
```

注意：上面两行加入的位置，加入错误将导致配置失败。下面以加粗斜体强调加入语句的位置。

查看 sysIACm-auth 文件显示如下：

```
authrequired pam_env.so  
authrequired pam_tally.so onerr=fail lock_time=60 deny=4  
authsufficient pam_unix.so unlllok try_first_pass  
authrequisiAC pam_succeed_if.so uid>=500 quiet  
authrequired pam_deny.so
```

```
account required pam_unix.so
```

```
account required pam_tally.so
```

```
...           ...           ...
```

下面是上面使用的选项的具体描述：

* onerr=fail

如果一些不可预料的事情发生，例如不能打开文件，这个决定了模块应该如何反应。

*lock_time=60

失败尝试后 60 秒拒绝登录，即每次登录需要间隔 60 秒。

* deny=4

这个选项表明如果这个用户登录 4 次失败，则这个用户将无法登录。

测试

切换到控制台：

用一个用户名登陆，输入 password 错误四次会锁定该用户。

有关其他的访问控制在此就不再详述。

1.7 系统安全管理

随着系统管理技能的日益提高，管理员会发现自己所关注的最大问题就是保持自己所在组织的网络及信息的安全性。合适的安全策略能够节省时间，并且根据系统中所存放内容的情况不同，甚至可以节省资金。为了保证系统的安全性，管理员必须理解所制定的安全性措施所保护的内容以及存在何种威胁，必须允许那些遵守系统使用规范的用户继续使用系统所提供的服务，同时将那些不遵守系统使用规则或者对系统安全具有危险性的用户排除在系统之外，同时还必须保证系统的数据与服务只能被那些具有访问权限的用户所访问，本章将从这些方面介绍中标麒麟安全操作系统软件 V7.0 的一些安全原则。

1.7.1 系统安全概要

中标麒麟安全操作系统安全 v5 包括下面几个方面的要素：物理安全管理、普通用户安全管理和超级用户安全管理。

1.7.1.1 物理安全

可以通过以下方面来消除物理安全：

- 保证放置计算机机房的安全，必要时需加报警系统，同时应提供软件备份方案，把备份好的软件放在安全的地点。
- 保证所有通信设施（包含有线通讯线、电话线、局域网和远程网等等）都不会被非法人员监听。
- 钥匙或信用卡识别设备、用户口令和钥匙分配、文件保护、备份或恢复方案等关键文档资料要保存在安全的位置。

1.7.1.2 普通用户安全管理

可以通过以下方面来消除普通用户安全：

- 系统管理员有责任发现并报告系统的安全问题，当普通用户登录时，其 shell 在给出提示前先执行/etc/profile 文件，要确保该文件中的 Path 指定做后搜索当前工作目录。
- 系统管理员可以定期抽取一个用户，将该用户安全检查结果（用户的登录情况简报、UID/GID 以及文件列表等等）发送到其部门即相关人员。
- 注意提高安全管理意识，系统管理员应强化安全规则，用户必须遵守个人安全标准，在权限允许范围内进行操作，也可以使用一些提高安全性的工具。

1.7.1.3 system 用户安全管理

可以通过以下方面来消除超级用户安全：

- 在日常使用中最好不要使用 root 账号，以普通用户进入系统可以防止对系统进行破坏性的操作，以 root 身份工作时应该保证输入的每个命令的正确性。
- 经常改变 root 的用户口令。
- 设置用户口令的时效。
- 不要把当前的工作目录排在 PATH 路径表的前面，以免特洛伊木马的侵入。
- 不要未注销账户就离开终端，特别是作为 root 用户时更不能这样。
- 可以将登录名 root 改成别的名称，使破坏者不能再 root 用户登录名下猜测各种可能的用户口令从而非法进入 root 账户。
- 查出不寻常的系统使用情况，如大量地占用磁盘、cpu 时间、进程，大量地使用 su 的企图，大量的无效登录到与某一个系统的网络传输以及可以的

uucp 请求。

- 保持系统文件安全的完整性，检查所有系统文件的存取许可，要特别注意设备文件的存取许可，任何具有 SUID 许可的程序都可能是黑客攻击的对象。
- 将磁盘的备份存放在安全的地方。
- 查出久未使用的登录账户，并取消此账户。
- 确保没有无用户口令的登录账户。
- 启动系统记账、加密等安全机制。
- 当安装来源不可靠的软件时，要检查源代码和 makefile 文件，查看特殊的子程序调用或命令。
- 如认为系统以泄密，就设法查出责任人与事故原因并及时进行补救。

1.8 资源利用

1.8.1 概述

系统资源是十分宝贵的。为了提高效率，高效的利用系统资源，中标麒麟安全操作系统软件 V7.0 采用了多种工具，通过不同的方式对系统资源进行合理的分配和利用。

1.8.2 示例

应通过一定措施确保当系统出现某些确定的故障情况时，SSF 也能维持正常运行，如系统应检测和报告系统的服务水平已降低到预先的最小值。要求资源（CPU 内存 磁盘）达到指定利用率，报警或提示都可以。

操作	描述
<p>1 root 用户启动 CPU 资源监控服务 cpumonitor:</p> <pre>/etc/init.d/cpumonitor start</pre> <p>2 运行一个大量运算的程序while [1] ; do echo ;done</p> <p>，当CPU占有率达到 99% 以上时，系统将弹出窗口提示用户。</p>	<p>CPU 占有率 99%以上时自动提示用户</p>

3 用户可以选择取消提示。	
---------------	--

应按资源分配中最大限额的要求，进行SSOOS资源的管理和分配，要求配额机制确保用户和主体将不会独占某种受控的资源。

操作	描述
<p>1 当某进程占用较多的CPU执行时间时，root用户使用nice调整其优先级。</p> <p>2 对指定进程设置优先级命令如下：（n为-20~20之间的整数，数值越大，优先级越低）。</p> <p><code>renice +n PID</code></p>	限制进程的优先级

系统应以每个用户或每个用户组为基础，提供一种机制，控制他们对磁盘的消耗和对CPU的使用

操作	描述
<p>1 root 用户开启选定分区（也可以是使用 dd 创建一个文件通过环设备挂载 <code>dd if=/dev/zero of=/mnt/tmp/quota.img bs=1024 count=1</code>）的磁盘限额功能，也可以通过在 <code>/etc/fstab</code> 文件中对应分区的 defaults 后面添加 <code>usrquota</code>，参考如下：</p> <p><code>/dev/sda1 /mnt ext3defaults , usrquota 1 1</code></p> <p>2 重新挂载选定的分区，并修改挂载点权限，可以使用如下命令：</p> <p><code>mount -o remount, rw, usrquota -t ext3 /mnt/tmp/quota.img /IACstquota</code></p> <p><code>chmod 777 /IACstquota</code></p> <p>3 检查分区使用情况：</p> <p><code>quotacheck -auv</code></p> <p>4 配置指定用户的磁盘限额，运行如下命令后在指定第一个 hard 对应的值，单位是 KB。</p> <p><code>edquota -u 用户名</code></p> <p>5 启用磁盘配额功能</p> <p><code>quotaon -a</code></p>	使用 quota 控制用户磁盘配额

6 当用户使用超过限定的磁盘大小时，系统即会报错（若限制大小为 10K，切换到该用户 执 行 <code>dd if=/dev/zero of=/mnt/tmp/quota.img bs=1024 count=9</code> 时就会报错）。	
---	--

操作	描述
<p>1.执行一个循环程序，使用 <code>top</code> 命令查看该进程的 <code>cpu</code> 使用情况。</p> <p>2.运行 <code>cpulimit -p 进程号 -l 10</code> #限制这个进程只能使用 10%的 CPU 资源。</p> <p>3.可以通过 <code>top</code> 看到这个进程确实最多占用 10%的 <code>cpu</code> 资源。</p>	使用 <code>cpulimit</code> 限制 <code>cpu</code> 的使用

操作	描述
<p>可以在 <code>/etc/pam.d/login</code> 中加入一行启用这个模块，如：</p> <pre>... Session include sysIACm-auth Session required pam_limits.so</pre> <p>例：限制用户 <code>user1</code> 创建单个文件大小不能大于 10000KB（值太小的话会无法登录）</p> <pre>User1 hard fsize 10000</pre> <p>使用这个用户登录后创建于 10000KB 的文件时就会报错。</p> <p>其他资源的限制同样可以在这个配置文件中设置。</p>	<p>使用 <code>pam_limits</code> 限制用户使用系统资源，<code>root</code> 用户同样受到限制</p> <p>限制的配置文件在 <code>/etc/security/limits.conf</code></p> <p>可以在配置文件中加入一行</p> <pre><domain> <type> <iIACm> <value></pre> <p><code>Domain</code> 可以为用户名或用户组名</p> <p><code>Type</code> 可以为软限制或硬限制</p> <p><code>IIACm</code> 可以为单个文件大小、总大小、进程数等</p> <p><code>Value</code> 为具体的值</p> <p>具体可以参考配置文件中的</p>

	注释。
--	-----

系统应提供软件及数据备份和复原的过程，在系统中应加入再启动的同步点，以便于系统复原

操作	描述
<p>1 root用户运行partimage 启动系统备份工具：</p> <p>partimage -z1 -o -d -b -V 10000 save /dev/sda1 备份文件名 -V 文件分段大小，-b 自动运行，-d 自动覆盖存在文件，save 备份，后面两个参数，第一为备份分区名，第二个为保存的备份文件名。操作需要管理员权限。</p>	系统备份功能测试

操作	描述
<p>运行partimage 还原系统，该步操作需要一个完整partimage系统备份文件：</p> <p>partimage restore /dev/sda1 备份文件</p>	<p>系统恢复功能测试</p> <p>操作需要管理员权限。</p> <p>（注意：操作将会覆盖掉指定分区数据）</p>

第 2 章 安全管理员

安全管理员负责整个系统的安全策略设计，配置与维护，以及在发现有造成潜在威胁时及时的发现并及时处理，保证系统安全。

2.1 双因子认证

双因子认证系统是一套高可靠的灵活定制的用户权限认证系统。功能分两部分：

1. 在用户指定情况下可以生成指定用户的双因子认证数据，并完成相关环境配置。
2. 对指定了使用双因子认证的用户在登录时进行用户权限验证。

双因子的权限管理是遵照三权分立系统规定的权限原则执行的，与传统的linux 中用户权限管理有一些差异。安全管理员可以操作任意非管理员用户的所

有身份鉴别信息；非安全管理员（包括系统管理员和审计管理员）可以完全自由操作自己的身份鉴别信息。普通用户可以在授权范围内操作自己的身份鉴别信息。

2.1.1 配置与使用

非安全管理员的配置：

(1) 在终端下运行：`douauth -i` 。如图 2-1 所示。

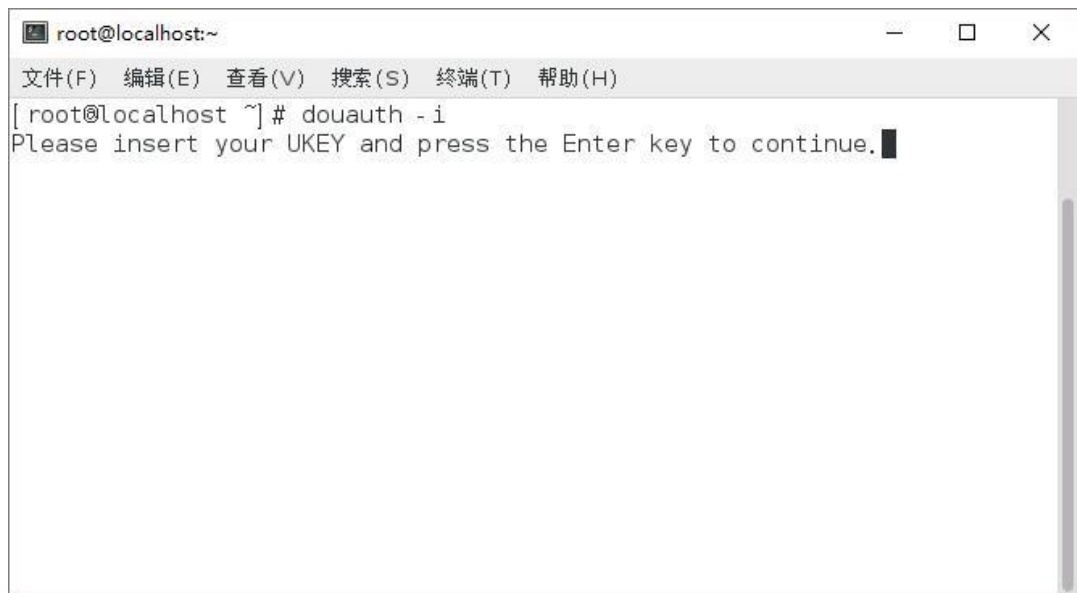


图 2-1 终端配置双因子认证

(2) 插入 UKEY，并按 Enter 键继续下步操作。如图 2-2 所示。

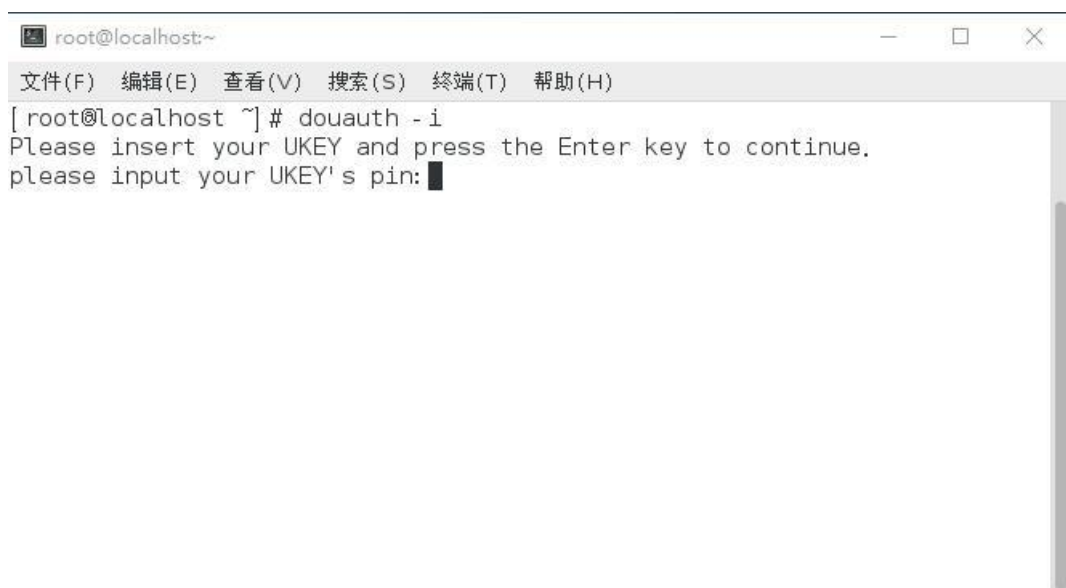


图 2-2 插入 UKEY

如果插入的 UKEY 与用户不对应或者输入了错误的 pin 码，则会提示错误。图 2-3，为输入错误的 pin 码时提示的错误。

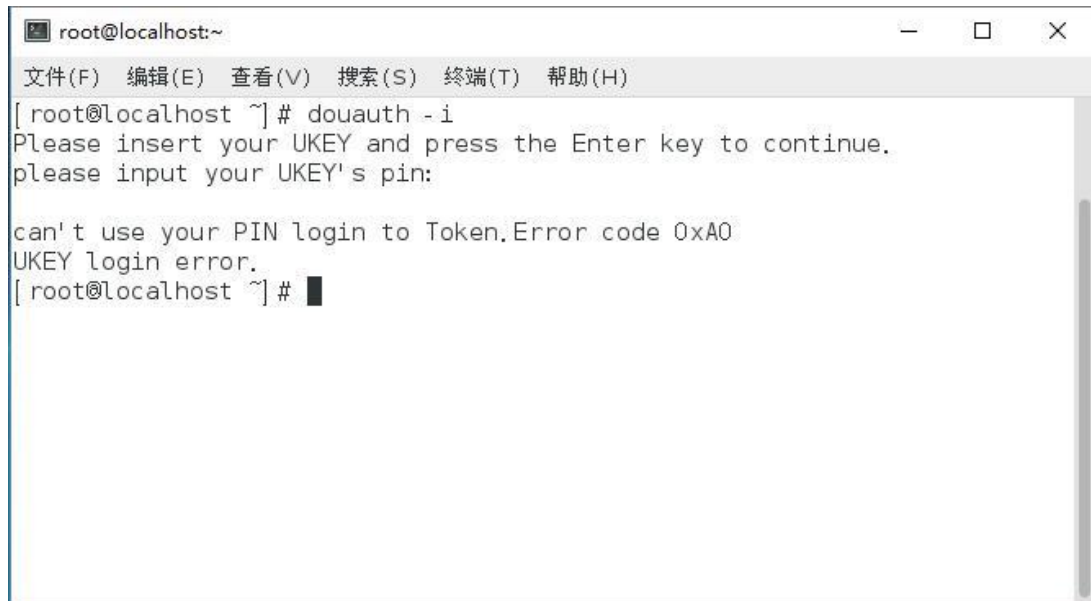


图 2-3 输入错误的 pin 码

(3) 插入正确的 UKEY，并输入正确的 pin 码，则可以按照提示开启或者关闭 UKEY，如图 2-4 所示。



图 2-4 开启或关闭 UKEY

(4) 如果需要开启 UKEY，则需要进行步骤 6。否则会提示用户 UKEY 认证功能被关闭，如图 2-5 所示。

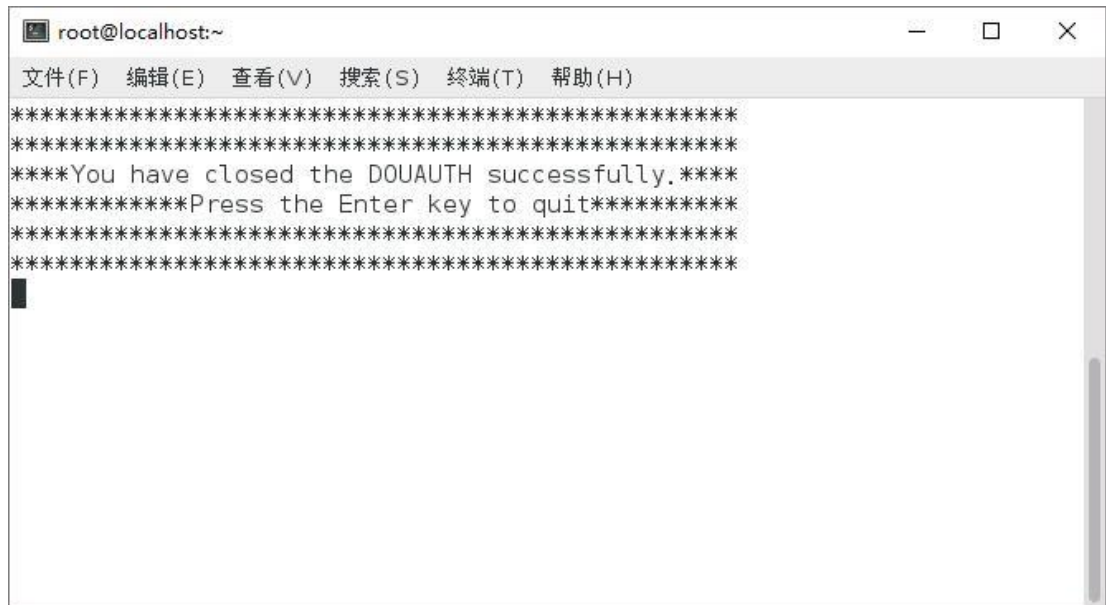


图 2-5 UKEY 已关闭

(5) 提示是否选择 UKEY 与系统同步的功能。如图 2-6 所示。

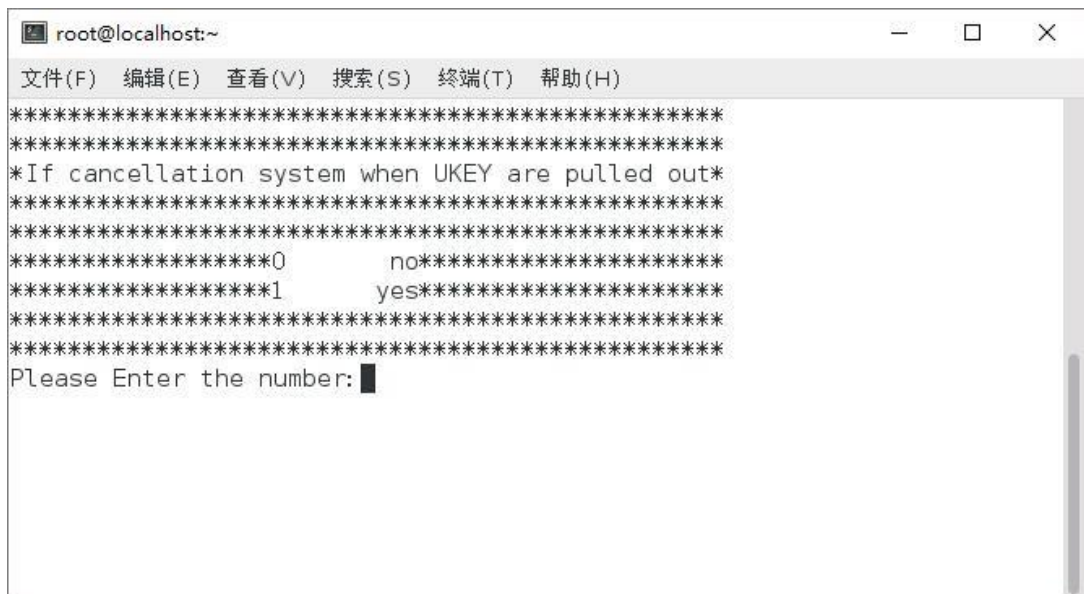


图 2-6 设置 UKEY 同步功能

(6) 最后，提示双因子认证开启成功，如图 2-7 所示。

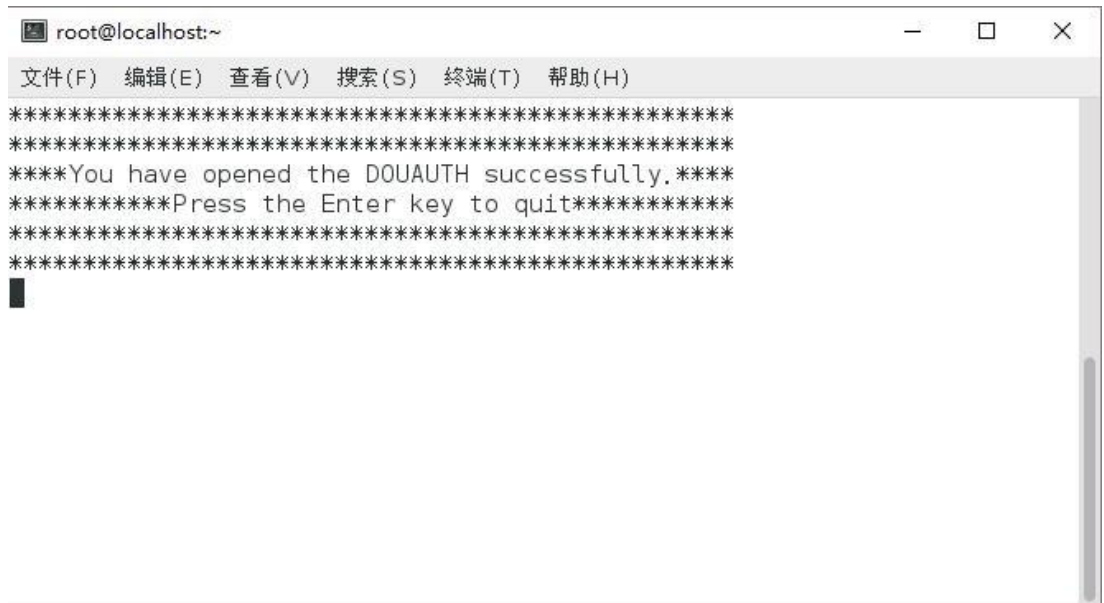


图 2-7 双因子认证设置成功

安全管理员配置：

(1) 安全管理员和非安全管理员的前三步配置是一样的，这里不再赘述。插入正确的 UKEY 并输入正确的 pin 码之后，提示用户选择配置安全管理员的双因子认证还是普通用户的双因子认证。如图 2-8 所示。

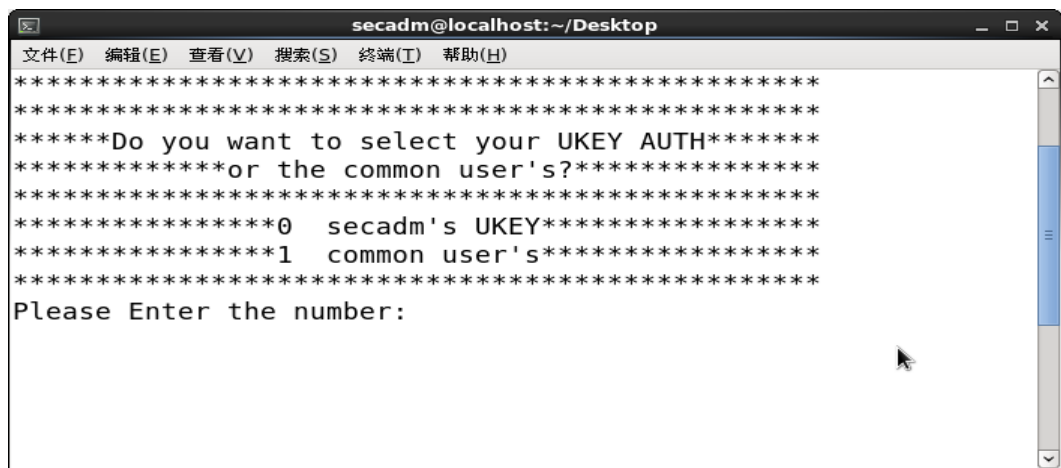


图 2-8 选择需要配置的用户

(2) 如果选择安全管理员的配置，则步骤和非安全管理员相同。否则，进入如图 2-9 界面，输入普通用户的用户名。



图 2-9 输入普通用户用户名

(3) 进入选择开启或关闭双因子认证界面，如图 2-4 所示。

(4) 如果选择关闭双因子，则如图 2-5 所示，成功关闭双因子认证。否则，需要选择是否强制配置双因子认证，如图 2-10 所示。

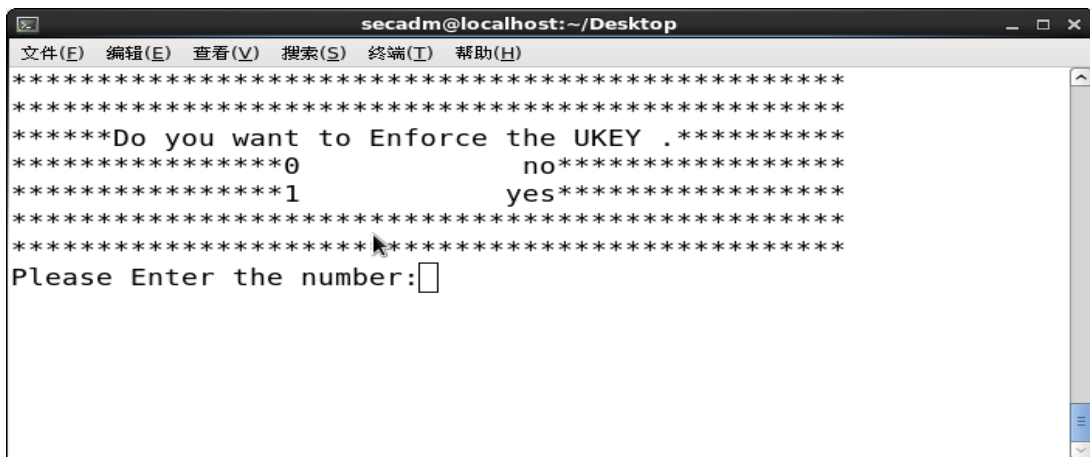


图 2-10 选择是否强制配置双因子认证

(5) 选择是否设置 UKEY 和系统同步，如图 2-6 所示。

(6) 提示开启双因子认证成功，如图 2-7 所示。

2.1.2 双因子认证使用

开启双因子认证的用户，开机时需要插入 USBKEY 设备进行权限验证，验证失败时将无法登录系统。

当系统设置为默认安全级别时，为了保证高安全性，双因子登录系统要求管理员必须启用双因子认证配置，否则拒绝任何管理员用户登录。管理员在进行模式切换操作时，如果有管理员没有启用双因子认证配置，系统将按一个预制的 UKEY 设备信息自动启用对应管理员用户的双因子认证配置。

2.1.3 注意

对特权用户使用双因子认证，要注意对认证设备和口令的保存，一旦丢失相关认证因子，将会导致系统无法登录。

2.2 CAP 管理

中标麒麟安全操作系统软件 V7.0 将系统管理员的权能进行了细分，表 2-8 给出了系统支持的 cap 属性：

能力名称	含义
CAP_CHOWN	进程进行 <code>chown()</code> 操作修改文件的属主 ID 和用户组 ID，覆盖进程属主 ID 等于文件属主 ID、进程用户组 ID 或进程附加组 ID 等于文件用户组 ID 的限制
CAP_DAC_OVERRIDE	进程执行文件时，覆盖文件访问权限位中对执行访问的限制
CAP_DAC_READ_SEARCH	进程读文件或搜索目录时，覆盖文件访问权限位中对读和搜索的限制
CAP_FOWNER	在 CAP_FSETID 未设置时，可以覆盖要求进程属主 ID 等于文件属主 ID 的文件操作限制
CAP_FSETID	当文件的 <code>S_ISUID</code> 和 <code>S_ISGID</code> 位被置上时，可以覆盖要求进程属主 ID 等于文件属主 ID 的文件操作限制；当文件的 <code>S_ISGID</code> 被置上时，可以覆盖要求进程所属组 ID 等于文件属主 ID 的文件操作限制
CAP_KILL	覆盖信号发送时要求发送进程的用户 ID/有效用户 ID 等于接收进程的用户 ID/有效用户 ID 的限制
CAP_SETGID	覆盖进程进行 <code>setgid()</code> 操作修改进程的真实用户组 ID 和只能修改有效用户组 ID 到真实用户组 ID 的限制；若系统支持保留 ID 时，覆盖只能修改保留的设置组 ID 为真实用户组 ID 或保留的设置组 ID 的限制
CAP_SETUID	覆盖进程进行 <code>setgid()</code> 操作修改进程的真实属主 ID 和只能修改有效属主 ID 到真实属主 ID 的限制；若系统支持保留 ID 时，覆盖修改保留的设置属主 ID 的限制

CAP_SETPCAP	允许设置当前用户 <code>permitIACd</code> 集合中的任何权能到任何进程；允许删除任何进程中属于当前用户 <code>permitIACd</code> 集合的任何权能
CAP_LINUX_IMMUTABLE	允许修改文件的 <code>S_IMMUTABLE</code> 和 <code>S_APPEND</code> 属性
CAP_NET_BIND_SERVICE	允许绑定到 1024 一下的 TCP/UDP sockets；允许绑定到 32 以下的 ATM VCI
CAP_NET_BROADCAST	允许广播消息和监听组播消息
CAP_NET_ADMIN	允许接口配置；允许 IP 防火墙，伪装和记账功能；允许设置 socket 调试选项；允许修改路由表；允许设置任意进程或进程组对 socket 的所有权；允许绑定到任何开放代理；允许设置服务类型；允许设置混杂模式；允许清除驱动统计信息；允许组播；允许读/写设备专用寄存器；允许激活 ATM 控制 socket
CAP_NET_RAW	允许使用 RAW socket;允许使用 PACKET socket
CAP_IPC_LOCK	允许对共享内存段加锁；允许 <code>mlock</code> 和 <code>mlockall</code>
CAP_IPC_OWNER	允许覆盖 IPC 所有权检测
CAP_SYS_MODULE	允许插入和删除内核模块，修改内核
CAP_SYS_RAWIO	允许 <code>ioperm/iopl</code> 访问；允许通过 <code>/proc/bus/usb</code> 发送 USB 消息到任何设备
CAP_SYS_CHROOT	允许使用 <code>chroot</code>
CAP_SYS_PTRACE	允许 <code>ptrace</code> 任何进程
CAP_SYS_PACCT	允许设置进程记账
CAP_SYS_ADMIN	允许配置安全密码；允许管理随机设备；允许检查和配置磁盘配额；允许配置内核 <code>syslog</code> （例如 <code>printk</code> 操作）；允许设置域名；允许设置主机名；允许调用 <code>bdflush()</code> ；允许 <code>mount()</code> 和 <code>umount()</code> ，设置新的 smb 连接；允许一些 <code>nfsservctl</code> ；允许 <code>VM86_REQUEST_IRQ</code> ；允许读/写 alpha 体系结构的 pci 配置；允许 mips 上的 <code>irix_prctl</code> ；允许 m68k 上的

	cache flush 操作；允许删除信号；允许加锁/解锁共享内存段；允许打开/关闭 swap；允许在块设备上设置 readahead 和缓冲 flush；允许在 socket 信任检测时伪造 pid；允许在软驱中设置几何；允许再 xd 驱动上开/关 DMA；允许管理 md 设备；允许调整 ide 驱动；允许访问 nvram 设备；允许管理 apm_bios、序列和 bttv 设备；允许 isdn CAPI 驱动的 manufacturer 命令；允许在 pci 配置空间读取非标准部分；允许在 sbpcd 驱动中 DDI 调试 ioctl；允许建立序列端口；允许发送 qic-117 命令；允许在 SCSI 控制器上打开/关闭标记队列，发送任意 SCSI 命令；允许在 loopback 文件系统中设置加密钥匙；允许设置 zone reclaim 策略
CAP_SYS_BOOT	允许使用 reboot
CAP_SYS_NICE	允许提升和设置其它进程的优先级；允许对自己的进程使用 FIFO 和 round-robin 调度；允许设置其它进程的调度算法；允许在其它进程上设置 cpu affinity
CAP_SYS_RESOURCE	允许覆盖资源限制，设置资源闲置；允许覆盖配额限制；允许为 ext2 文件系统保留空间；允许在 ext3 文件系统修改数据 journaling 模式；允许覆盖 IPC 消息队列的长度限制；允许使用实际时钟中大于 64hz 的中断；允许覆盖控制台分配的最大数量；允许覆盖 keymap 的最大数量；
CAP_SYS_TIME	允许操作系统时钟；允许在 mips 上 irix_stime；允许设置实际时钟
CAP_SYS_TTY_CONFIG	允许配置 tty 设备，允许 tty 上的 vhangup ()
CAP_MKNOD	允许 mknod () 的特权
CAP_LEASE	允许对文件采用 lease 操作

2.3 MLS 管理和 IAC 管理

MLS 策略实现了基于 BLP 模型的多级安全策略。MLS 策略下进程主体与客体的安全标记包含一个安全级。每个安全级由安全级别、安全类别组成：安全级别是从 1 到 256 的整数；安全类别则是一个 65536 个元素组成的集合的子集；同

时系统中定义三个特殊的安全标记：**low**、**high** 和 **public**。安全级中可不包含安全类别集，表示安全类别集为空。IAC 策略实现了基于 BIBA 模型的多级安全策略。BIBA 策略下主体（进程等）的完整性等级。

2.3.1 命令集

1. isic_get

用于获取进程、文件、用户的安全级和完整性，命令格式如下：

```
isic_get{-f|--file} [filename]
```

```
        {-p|--process} [pid]
```

```
        {-u|--uid} [uid]
```

示例：

- 获取文件/tmp/file1 的安全级

```
isic_get -f /tmp/file1
```

- 获取用户 root 的安全级

```
isic_get -u root
```

- 获取进程的安全级， 首先使用 `ps aux | grep [argument]` 获取进程号，再使用

```
isic_get -p [pid]
```

2. isic_set

设置文件、用户、进程的安全级和完整性等级，命令格式如下，

```
isic_set -p [pid] -m MLS_TYPE_PUBLIC -i [1]
```

```
isic_set -f [filename] -m [MLS_TYPE_PUBLIC] -i [2]
```

```
isic_set -u [uid] -m [MLS_TYPE_PUBLIC] -i [3]
```

示例：

- 设定文件的安全级

```
isic_set -f /tmp/file1 -m g3:c2+c3 -i 1
```

- 设定进程的安全级

```
isic_set -p 123456 -m g3:c2+c3 -i 2
```

- 设定用户的安全级

```
isic_set -u 1005 -m g3:c2+c3 -i 3
```

3. isic_init

初始化文件、用户、进程的安全级，命令格式如下：

```
isic_init {-f | --filename} [filename] [mls_patIACrn]
```

```
isic_init {-d | --directory} [dir] [mls_patIACrn]
```

```
isic_init {-a | -all} -f files //初始化所有磁盘文件
```

```
isic_init {-a | -all} -u users //初始化系统中 login 用户的所有安全级
```

示例:

- 初始化文件的安全级

```
isic_init -f /tmp/test type_public
```

- 初始化目录的安全级

```
isic_init -d /tmp/ type_public
```

- 初始化所有用户

```
isic_init -a users
```

- 初始化所有磁盘文件

```
isic_init -a files
```

2.4 ACL 使用

2.4.1 ACL 概述

文件 ACL 是客体的一个 DAC 实体（这里，ACL 不是指相应的访问控制机制，而是指一个实体，下同），包含一个项目列表，其中每一项是标识符（如：用户或用户组）和相应访问许可权集合。

ACL 有访问 ACL（access ACL）和缺省 ACL（default ACL）两种。访问 ACL

可以和任何文件系统对象关联，控制对它们的访问许可权。缺省 ACL 仅仅和目录关联，在包含它的目录下创建的非目录对象将把它继承为访问 ACL。在缺省的情况下（即未对文件的 ACL 进行过设置），由文件访问权限模式（mode）中 3 组 9 位（owner/group/other）直接转换而得到文件的基本 ACL 表，共有三个基本 ACL 项：基本 user、基本 group 及 other，基本 ACL 表项与文件访问权限模式总是保持一致，修改其中之一将会导致另一项的改变；此外，文件还可能具有附加 ACL 项（additional entries）。附加 ACL 项用于定义系统中除文件属主及属组以外其它用户或用户组所具有的对文件的访问权限，还可以定义 ACL 掩码。

2.4.2 相关命令

本节介绍所有用户都可以使用的 ACL 命令，包括 getfacl 和 setfacl。文件的属主和系统管理员可以设置、获取文件的 ACL。

2.4.2.1 获取文件 ACL

getfacl 命令用于获取文件 ACL，命令格式如下：

```
getfacl [-dh] filename
```

此命令将显示文件 filename 的 ACL。任选项 -d 表示显示文件的目录缺省 ACL（仅当 filename 是一个目录时有效），而不是文件的访问 ACL；-h 表示当 filename 是一个符号连接时，显示该符号连接的访问 ACL，而不是符号连接所连接文件的 ACL。

以下是 getfacl 命令的一些使用实例，其中加粗部分为系统显示的内容，“/****”到 “****/” 之间为文档编写者进行的注释。

```
[root@wlf IACst]# ls -l
total 2
-rw-rw-r-- 1 root root 0 5月 9 01:50 a
lrwxr-xr-x 1 root root 1 5月 9 01:50 a2 -> a
[root@wlf IACst]# getfacl a
#file:a
#owner:0
#group:0
user::rw-
user:wlf:rw-
```

```
user:zly:r--
group::r--
mask::rw-
other::---
[root@wlf IACst]# getfacl a2
#file:a2
#owner:0
#group:0
user::rw-
user:wlf:rw-
user:zly:r--
group::r--
mask::rw-
other::---
[root@wlf IACst]# getfacl -h a2
#file:a2
#owner:0
#group:0
user::rwx
user:wlf:rw-          # effective: r--
group::r--
mask::r-x
other::r-x
```

/*** 上述第一条命令显示当前目录下有两个文件 a 和 a2，其中 a2 是 a 的符号连接；第二条命令获得文件 a 的访问 ACL，显示表明文件属主对文件可读可写，用户 wlf 对文件可读可写，用户 zly 对文件只读，文件属主用户组其它用户对文件只读，文件访问掩码为可读可写，其它用户不可访问该文件；第三条命令显示 a2 的 ACL，实际上也是 a 的 ACL；第四条命令显示符号连接 a2 的 ACL，显示表明文件属主对文件可读可写可搜索该符号连接，用户 wlf 对符号连接可读可写（但实际上只读），文件属主用户组其它用户对符号连接只读，文件访问掩码为可读可搜索，其它用户可读可搜索该符号连接。 */

```
[root@wlf /root]# ls -ld IACst
drwx---r-x    2 root    root          512  5 月  9 02:18 IACst
[root@wlf /root]# getfacl IACst
#file:IACst
```



```
#owner:0
#group:0
user::rwx
user:wlf:rwx
user:zly:r--
group::r-x
group:ttt:---
mask::rwx
other::r-x

[root@wlf /root]# getfacl -d IACst
#file:IACst
#owner:0
#group:0
user::r--
user:wlf:rw-
group::r--
mask::rw-
other::---

[root@wlf /root]# cd IACst
[root@wlf IACst]# touch file1
[root@wlf IACst]# getfacl file1
#file:file1
#owner:0
#group:0
user::rw-
user:wlf:rw-          # effective: r--
group::r--
mask::r--
other::r--
```

/* ** 上述第一条命令显示当前目录下有一个名为 IACst 的目录；第二条命令显示该目录的访问 ACL，结果表明目录属主对目录可读可写可搜索，用户 wlf 对目录可读可写可搜索，用户 zly 对目录只读，目录属主用户组中其它用户对目录可读可搜索，用户组 ttt 中的用户对目录不可访问，其它用户对目录可读可搜索；第三条命令显示目录的缺省 ACL，它对该目录下文件的 ACL 有影响；第五条命令在该目录下创建文件 file1；第六条命令显示该文件的访问 ACL，结果表明文件自动

获得了关于用户 wlf 的 ACL 项。 ***/

2.4.2.2 设置文件 ACL

setfacl 命令负责设置文件 ACL，命令格式如下：

```
setfacl [-bdhkn] [-m entries] [-M file1] [-x entries] [-X file2]
[file ...]
```

命令将设置文件 file 的 ACL。命令中各任选项的含义如下：

- b 删除所有的 ACL 附加入口项，除了文件属主、文件属主用户组和其它用户三个必须的 ACL 基本入口项；如果删除的 ACL 入口项中包括“mask”入口项，那么保留的文件属主用户组 ACL 入口项为原有的文件属主用户组 ACL 入口项与 mask 入口项计算后的有效值；
- d 仅当 file 为目录时有效，表示设置目录的缺省 ACL；
- h 仅当 file 为符号连接时有效，表示设置符号连接本身的 ACL；
- k 删除缺省 ACL；
- m entries 按照 entries 的定义增加文件的 ACL 入口项，一般 entries 使用 ACL 的短文本表示形式；
- M file1 按照文件 file1 中的定义增加文件的 ACL 入口项，一般 file1 种使用 ACL 的长文本表示形式；
- n 增加 ACL 入口项后，不计算访问掩码；
- x entries 按照 entries 的定义删除文件的 ACL 入口项，一般 entries 使用 ACL 的短文本表示形式；
- X file2 按照文件 file2 中的定义删除文件的 ACL 入口项，一般 file2 种使用 ACL 的长文本表示形式；

以下是 setfacl 命令的一些使用实例，其中加粗部分为系统显示的内容，“/***”到“***/”之间为文档编写者进行的注释。

```
[root@wlf IACst]# ls -l a
-rwxrwxr-x    1 root    root          0  5 月  9 03:26 a

[root@wlf IACst]# setfacl -m u::rwx, g::r-x, o::--- a
[root@wlf IACst]# ls -l a
total 6
-rwxr-x---    1 root    root          0  5 月  9 03:27 a

[root@wlf IACst]# getfacl a
#file:a
#owner:0
```

```
#group:0
user::rwx
group::r-x
mask::r-x
other:---
```

/** 第一条命令显示了文件 a 的文件访问权限；第二条命令设置了文件 a 的基本 ACL 表项；第三条命令再次显示文件 a 的访问权限，结果表明文件访问权限因为文件基本 ACL 的修改而发生改变，并且文件访问权限与设置的文件基本 ACL 表项一致；第四条命令显示文件的 ACL 表，结果表明与设置一致 */

```
[root@wlf IACst]# setfacl -m u:user1:rwx, g:group1:rw- a
[root@wlf IACst]# getfacl a
#file:a
#owner:0
#group:0
user::rwx
user:user1:rwx
group::r-x
group:group1:rw-
mask::rwx
other:---
```

/** 在前面命令的基础上，继续执行，第一条命令修改文件 a 的 ACL，增加了用户 user1、用户组 group1 的 ACL 入口项；第二条命令显示文件的 ACL 表，结果表明对应用户 user1、用户组 group1 的 ACL 入口项已增加，并且与设置相同。 */

```
[root@wlf IACst]# setfacl -m m::r-- a
[root@wlf IACst]# getfacl a
#file:a
#owner:0
#group:0
user::rwx
user:user1:rwx          # effective: r--
group::r-x              # effective: r--
group:group1:rw-        # effective: r--
mask::r--
```

```
other::---
```

/** 在前面命令的基础上，继续执行，第一条命令修改文件 a 的访问掩码 mask 为只读；第二条命令显示文件的 ACL 表，结果表明掩码设置成功，并且由于掩码的设置，对应用户 user1、属主用户组、用户组 group1 的 ACL 入口项的有效值也发生了相应的变化。 */

```
[root@wlf IACst]# setfacl -x g:group1:rw- a
```

```
[root@wlf IACst]# getfacl a
```

```
#file:a
```

```
#owner:0
```

```
#group:0
```

```
user::rwx
```

```
user:user1:rwx          # effective: r--
```

```
group::r-x              # effective: r--
```

```
mask::r--
```

```
other::---
```

/** 在前面命令的基础上，继续执行，第一条命令删除文件 a 的关于用户组 group1 的 ACL 入口项；第二条命令显示文件的 ACL 表，结果对应用户组 group1 的 ACL 入口项被成功删除。 */

```
[root@wlf IACst]# setfacl -b a
```

```
[root@wlf IACst]# getfacl a
```

```
#file:a
```

```
#owner:0
```

```
#group:0
```

```
user::rwx
```

```
group::r-x
```

```
mask::r-x
```

```
other::---
```

/** 在前面命令的基础上，继续执行，第一条命令删除文件 a 的所有附加 ACL 入口项；第二条命令显示文件的 ACL 表，结果表明删除成功。 */

```
[root@wlf /root]setfacl -d -m u::rw, g::r, o::r, m::r IACst/
```

```
[root@wlf /root]getfacl -d IACst/
```

```
#file:IACst/
```

```
#owner:0
```

```
#group:0
```

```
user::rw-
group::r--
mask::r--
other::---
[root@wlf /root]setfacl -d -n -m u:wlf:rwx IACst/
[root@wlf /root]getfacl -d IACst/
#file:IACst/
#owner:0
#group:0
user::rw-
user:wlf:rwx          #effective: r--
group::r--
mask::r--
other::---
```

/** 第一条命令设置目录 IACst 的缺省 ACL，第二条命令显示目录的缺省 ACL，第三条命令为目录的缺省 ACL 添加一项为用户 wlf 设置权限的 ACL。注意：在设置目录的缺省 ACL 时，如果目录没有缺省 ACL，则必须首先添加基于目录属主、属组以及其他的缺省 ACL，然后才能添加为其他用户或组授权的 ACL。 **/

2.5 系统注意事项

在系统使用过程中，安全管理员在进行安全属性设置及用户角色权限等管理时，应注意不要为某个角色设置超出其实际应该使用的权限，否则会造成安全隐患。

一旦因安全配置原因导致系统不能正常使用，可以进入系统维护模式进行维护。

第 3 章 审计管理员

对系统中所有的操作事件进行审计，并将审计的结果进行整理分析，定期保存审计日志，并对日志中的可疑事件提交 secadm 处理。

3.1 安全审计

审计（audit）是 Linux 系统中保障系统安全的一个重要组件。审计服务能实时、全面地记录对文件、文件夹、系统资源的访问、系统调用情况。通过指定有效的审计规则，所有对系统安全存在风险的事件都会被记录在案。方便审计管理员事先的防范和事后的调查取证。安全控制中心的安全审计模块提供图形化的界面，协助操作系统审计管理员查看、查询相关审计日志、审计报表，设计审计规则，提供系统告警服务等。

安全审计与安全数据库系统结合，提供进程级独立安全审计功能，包括提供审计日志、实时报警生成，潜在侵害分析、基于异常检测，基本审计查阅、有限审计查阅和可选审计查阅，安全审计事件选择，受保护的审计踪迹存储，审计数据的可用性确保，审计数据的安全备份以及审计数据的访问控制等功能。

审计相关术语如下：

审计进程：审计进程是指由 `auditd` 命令运行后产生的 `auditd` 无交互后台守护进程（`audit daemon`）。

审计配置文件：审计服务的配置文件是指 `/etc/audit/auditd.conf`。

审计规则文件：审计规则文件是指 `/etc/audit/audit.rules`。该文件以命令参数的形式记录审计规则，审计进程启动时会读取该文件自动加载。管理员也可以在审计进程启动后，动态添加审计规则。

审计日志文件：审计日志文件是指 `/var/log/audit/audit.log`。它以文本形式记录下匹配审计规则的所有事件。用户也可以在审计配置文件中另外指定日志文件的位置。

3.2 使用说明

安全审计包含两个部分：系统审计管理和系统告警管理。

安装好中标麒麟安全操作系统软件 V7.0 以后，点击系统的启动项，选择【应用程序】→【系统工具】→【中标麒麟安全控制中心】→【系统安全】→【安全审计】弹出如下图所示界面：



图 3-1 开启安全审计界面

单击【开启安全审计】，出现如下图所示界面：



图 3-2 安全审计主界面

此界面是安全审计的主界面，可以看到安全审计主要有两大功能块：系统审计管理和系统告警管理。


3.3 系统审计管理

系统审计管理功能块分为三个部分：审计信息、审计报表和审计规则。如图 3-3 所示。



图 3-3 审计信息界面

3.3.1 审计信息

审计信息模块显示如图 3-3 所示。若选中全选，则表格第一列的复选框都被选中，若点击取消，则表格第一列的复选框全被取消选中。单击界面左侧的  按钮，可以进行信息筛选。审计类型可以选择要查询的审计类型，审计类型有三种：登录信息、AVC 和 CTMM，可以选择一种类型，也可以选择两种或多种类型查询。时间区间有五种：当天、最近一周、最近两周、最近一个月、最近三个月。此外，您还可以设置审计内容和审计时间，点击查询按钮，则表格显示输入查询条件后的审计内容，查询条件有审计类型、审计时间和审计内容。若直接点击删除按钮，则界面提示请选择需要删除的行，如图 3-5 所示，若选择需要删除行的

复选框，则界面提示您已删除选中的行，可以选择多行同时删除，如图 3-6 所示，删除的数据不仅在表格里面删除，数据库也同时删除。



图 3-4 带有高级查询的审计信息界面



图 3-5 删除提示

对表格的每行数据点击右键，显示查看详情和删除两个功能键，点击删除，则此行数据被删除。点击查看详情，弹出显示此行数据详细信息的对话框，如图 3-7 所示，此对话框显示当前点击行的审计时间、审计类型和审计内容，点击下一条，相应的显示框显示下一条表格数据，同时主界面表格相应的下一条数据变

色，如图 3-8 所示。若已经是最后一条数据，则提示已经是最后一条数据，点击上一条类推，若已经是第一条数据，则提示已经是第一条数据。

鼠标浮上表格的每一行数据，会出现一个框，显示此行的审计内容。

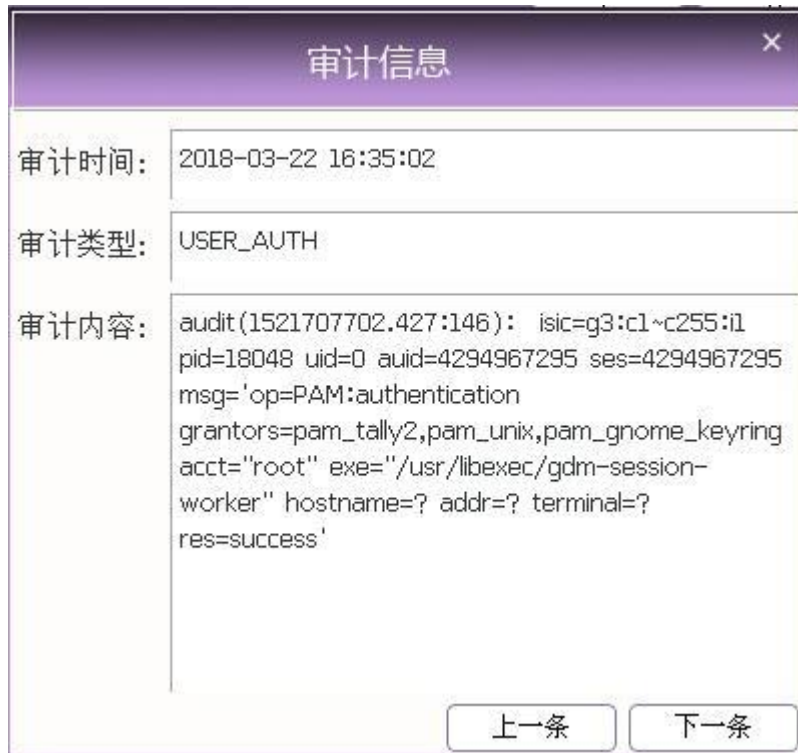


图 3-7 查看详情对话框



图 3-8 点击下一条界面显示

3.3.2 审计报表

点击审计报表，出现如图 3-9 和图 3-10 所示界面，这是登录信息、AVC 信息和 CTMM 信息的图表统计显示方式。

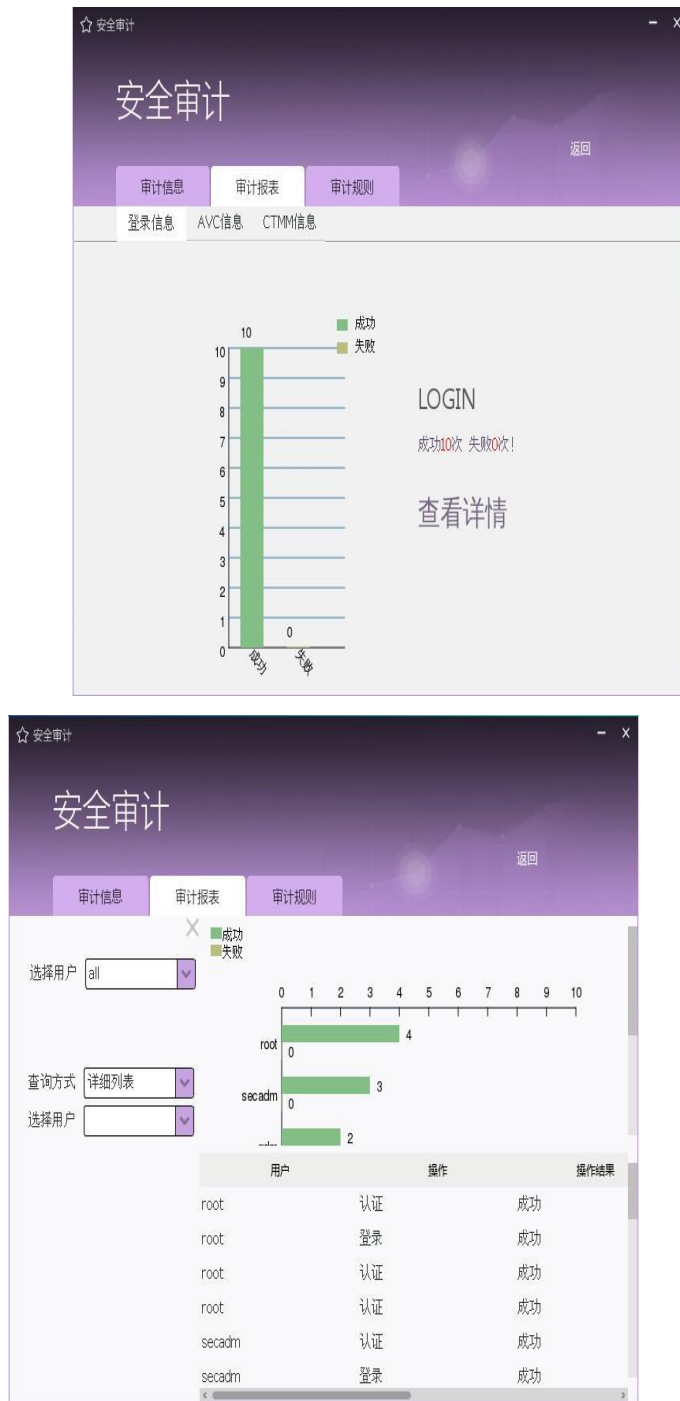


图 3-9 登录信息信息显示

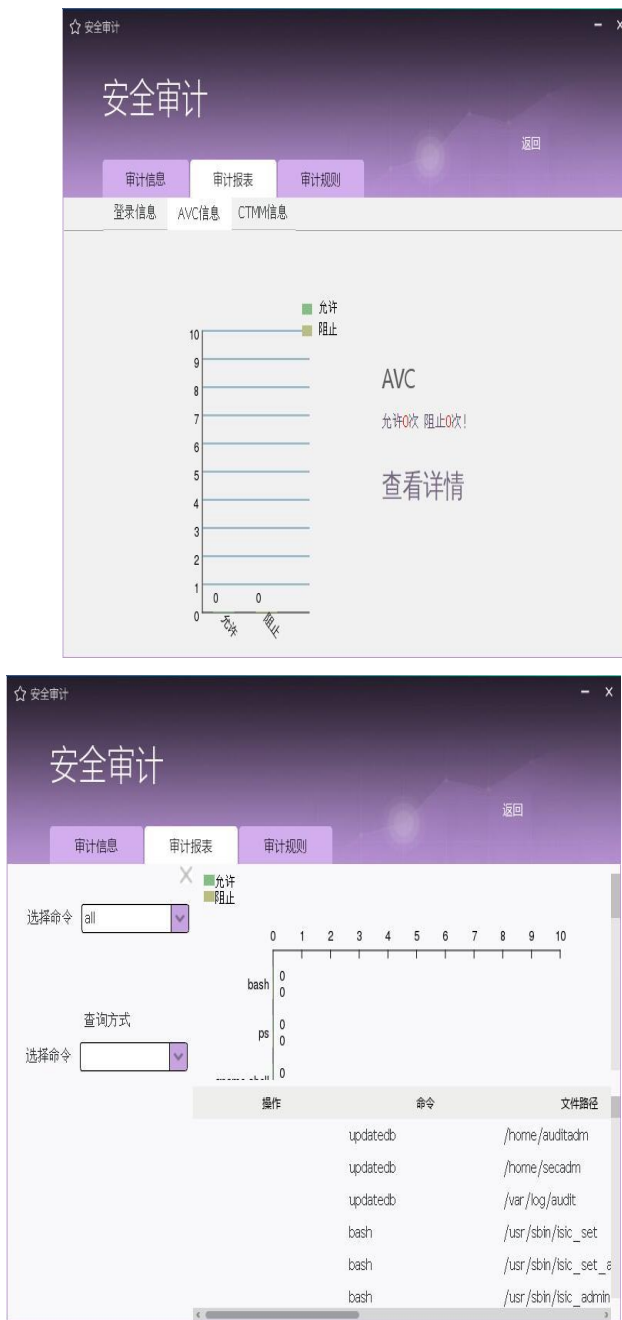


图 3-10 AVC 信息显示



图 3-11 CTMM 信息显示

如图 3-9 左侧登录信息显示，绿色代表登录成功，红色代表登录失败，点击查看详情，页面跳转到如图 3-9 右侧。点击【查询设置】，可以选择用户查看相应的用户的登录信息；查询方式可以按详细列表或时间列表。图右侧的上部分为图示展示，下部分为表格展示。

如图 3-10 左侧 AVC 信息显示，绿色代表允许，红色代表阻止。点击查看详情，界面跳转到如图 3-10 右侧所示。点击【查询设置】，可以选择命令；查询方式也可以选择命令。图右侧的上部分为图示展示，下部分为表格展示。

如图 3-11 左右 CTMM 信息显示，紫色代表允许，绿色代表阻止，黄绿色代表度量成功，灰土色代表度量失败。点击查看详情，页面跳转到如图 3-11 右侧所示页面。点击【查询设置】，可以选择要查询的文件；查询方式也可以选择文

件。图右侧的上部分为图示展示，下部分为表格展示。

3.3.3 审计规则

点击审计规则选项，界面显示如图 3-18 所示。



图 3-18 审计规则界面显示

如图所示，审计规则设置有五个选项：监控目录设置、监控文件设置、监控用户设置、监控进程设置和过滤日志信息。点击审计规则选项，界面默认选中监控目录设置框，用户可以对想要监控的目录进行设置，如点击右边的图标选中想要监控的目录，并可以对目录的权限进行选择，读、写、执行或者修改，可以输入关键字标识，设置好以后，点击应用规则，则设置好的信息会显示在下面的表格中，设置成功以后，界面会提示审计规则设置成功，如图 3-19 所示。规则设置成功以后，被监控的目录如果有任何操作，都会被记录在审计日志文件 audit.log 中。对表格中的每行点击右键，显示删除操作，点击删除，可以将该行数据删除。点击清空规则，表格中的数据将被清空，同时审计日志文件里添加的这条规则将被删除。



图 3-19 审计规则设置成功

若监控目录为空，界面提示监控路径不能为空，如图 3-20 所示。



图 3-20 监控目录信息页面

选择监控设置项可以通过点击左边的目录选择，也可以通过滑动滚动条选择。设置完监控目录以后，可以设置监控文件，原理与设置监控目录是一样的。

监控进程设置界面如图 3-22 所示，输入所要监控的进程名，点击应用规则按钮，表格里插入一条进程数据。规则设置成功以后，审计日志文件里面就会有监控此项进程的日志。

过滤日志信息界面如图 3-23 所示，输入要过滤的日志类型，如 SYSCALL、USER_LOGIN，点击应用规则，下面的表格插入一条过滤日志类型数据。规则设置成功以后，审计日志文件里面就不会有此类型的日志信息。



图 3-22 监控进程信息页面



图 3-23 过滤日志信息页面

3.4 系统告警管理

系统告警管理模块分为两个部分：告警信息和告警设置。

3.4.1 告警信息

告警信息部分主要显示系统设置的告警信息，界面如图 3-24 所示。选中全选按钮，表格里面的复选框可以全部被选中，再点击删除按钮，则表格里面的数据全部被删除，同时数据库里的数据也全部被删除。类别控件里面有三个可以选择的类别：LOGIN、AVC 和 CTMM，可以选择一个，也可以同时选择多个。状态控件有未查看和已查看两种状态，可以选择一个，也可以同时选择两个。点击查询按钮，表格将显示输入类别和状态条件以后的查询结果。点击删除按钮，界面会提示请选择需要删除的行，若选中将要删除的行，则界面提示您已经删除所选的行（跟审计信息界面类似，不再显示图片）。刷新按钮的功能是：若系统产生了一条告警信息，则点击刷新按钮，表格里面添加一条告警信息。时间区间和高级设置的功能类似审计信息页面，不再赘述。



图 3-24 告警信息页面

对表格内的每行点击右键，显示查看详情、标记为未查看或者已查看和删除三个快捷键，若当前行的状态为未查看，则快捷键显示为已查看，反之，则显示未查看，查看详情和删除的功能类似审计信息页面，不再赘述。

3.4.2 告警设置

告警设置部分主要设置告警方式和告警条件，界面如图 3-25 所示。



图 3-25 告警设置页面

告警方式设置主要分为消息发送告警设置和邮件发送告警设置，界面如图 3-25 所示。消息发送告警设置有两个选项：是否发送系统消息和发送方式，用户可以自行勾选所需要的项，若勾选，则以选择的发送方式发送系统消息。邮件发送告警设置五个选项：是否发送邮件、发件服务器地址、帐号、密码和收件人邮箱。若勾选发送邮件，则把下面四项输入，点击保存，就可以发送告警信息到收件人邮箱里面。

告警条件设置主要分为登录信息设置、AVC 信息设置和 CTMM 信息设置。登录信息设置界面如图 3-26 所示。由图中可看出，登录信息设置有六个选项：是否发送登录告警、登录时间段、连续登录失败次数、用户登录频率、登录 IP 和超级用户登录告警。若勾选发送登录告警并选择登录时间段，则用户在选择的时段里面登录系统，系统就会收到登录告警信息，如图 3-27 所示。AVC 信息设置界面如图 3-28 所示。AVC 信息设置有四个选项：是否发送 AVC 告警、拒绝告警、危险操作和策略匹配。若勾选发送 AVC 告警，并有危险操作如 setenforce 操作，则系统发送告警信息，点击表格右边的添加按钮，表格可以添加策略，点

击删除按钮，则表格里面的数据可以删除。



☆ 安全审计

安全审计

保存 取消 返回

告警信息 告警设置

消息发送 邮件发送 登录信息 AVC信息 CTMM信息

是否发送登录告警 ☒

登录时间段

☐ 00:00~02:59 ☐ 03:00~05:59

☐ 06:00~08:59 ☐ 09:00~11:59

☐ 12:00~14:59 ☐ 15:00~17:59

☒ 18:00~20:59 ☒ 21:00~23:59

连续登录失败次数 大于 5 次

用户登录频率 10分钟内 大于 10次

登录IP ☐ 局域网IP ☒ 外网IP

获取超级用户权限告警 ☒

图 3-26 登录信息设置

☆ 安全审计

安全审计

保存 取消 返回

告警信息 告警设置

消息发送 邮件发送 登录信息 AVC信息 CTMM信息

是否发送AVC告警 ☒

拒绝告警 ☐

危险操作 ☒ setenforce

策略匹配

源类型	目标类型	tclass
system_t	security_t	security
secadm_t	boot_t	file

添加 删除

图 3-28 AVC 信息设置页面

☆ 安全审计

安全审计

保存 取消 返回

告警信息 告警设置

消息发送 邮件发送 登录信息 AVC信息 CTMM信息

是否发送CTMM告警 ☒

拒绝告警 ☐

度量失败告警 ☐

图 3-29 CTMM 信息设置页面

CTMM 信息设置页面如图 3-29 所示，CTMM 信息设置共有三个选项：是否获取 CTMM 告警、拒绝 CTMM 告警和度量失败告警。若勾选获取 CTMM 告警和度量失败告警，则系统接收 CTMM 告警和度量失败告警。