PAM认证机制

1 PAM认证机制简介

为安全起见，计算机系统只冇经过授权的合法用户才能访问，在这里如何正确鉴别用户的真实身份是一个关键的问题。所谓用户鉴别，就是用户向系统以一种安金的方式提交口己的身份证明，然后山系统确认用八的身份是否属实的过程。换句话说，用八鉴别是系统的门户，每个用户进入到系统中都必须经过鉴别这一道关。

嵌入式认证模块(PAM)机制采用模块化设计和插件功能，使得我们可以轻易地在应用程序中插入新的鉴别模块或替换原先的组件，而不必对应川程序做任何修改，从而使软件的定制、维持和升级更加轻松，因为鉴别机制与应用程序Z间相对独立。应用程序对以通过PAM API方便的使用PAM提供的各种鉴别功能，而不必了解太多的底层细节。

此外，PAM的易用性也较强，主要表现在它对上层屏蔽了鉴别的具体细节，所以用户不必被迫学习各种各样的鉴别方式，也不必记住多个口令;又由丁•它实现了多鉴别机制的集成问题,所以单个程序可以轻易集成多种鉴别机制如Kerberos鉴别机制和Diffie-Hellman鉴别机制等，但用户仍町以用同一个口令登录而感觉不到采取了各种不同鉴别方法。

在广人开发人员的努力下，各版木的UNIX系统陆续提供对PAM的支持。其中，Linux-PAM(Pluggable Authentication Modules for Linux)是专门为 Linux 操作系统实现的， 包括 Debian Linux 2.2> Turbo Linux 3.6、Red Hat Linux 5.0 以及 SuSE Linux6.2 及它们的后续版木都提供对PAM的支持。FreeBSD从3.1版开始支持PAM。需要注意的思 除了具休实现不同外，各种版本Unix系统上的PAM的框架是相同的，所以本文介绍的 Linux-PAM框架知识具有普遍性。

2 Linux-PAM 的配置

Linux-PAM的目标就是为系统管理者提供最大限度的灵活性。系统管理者可以通过两种形式对Linux-PAM进彳亍配遇：单•配登文件/etc/pam.conf;或者是/etc/pam.d/H录。I、■面我们将讨论其配置文件的语法，接着给出一些实际应用的例子，以供读者参考。

2.1 Linux-PAM单一配置文件的语法

通过图1读者可能会注意到，配查文件也放在了在应用接口层小，它与PAM API配合使用，从而达到了在应用中灵活插入所需鉴别模块的日的。它的作用上要是为应用选定具体的鉴别模块，模块间的组合以及规定模块的行为。

在使用该配置文件前，读者首先应该明白Linux-PAM的记号是大小写敏感的。有两个特殊的符号：“#”和“.”。配置文件中的注释以#开头，一般配置文件中每行是一个入口（除注释外），但是如果某个入口的定义很长，可以通过使用转义符回行，而下一行也被看作是这个入口的一部分。

一般/etc/pam.conf文件每行的格式如下：

service-name module-type control-flag module-path arguments

其中，每个字符段的具体含义如下：

service-name： 为这个入口分配的服务名。通常这是给定应用程序的会话名。例如：ftpd^ rlogind> su等等。Linux-PAM还为默认的验证机制保留一个特殊的服务名，就是ohter,人小写均可。另外，如果某个模块指定了以命名的服务，那other就应该被忽略。

modle-type： Linux-PAM当前有四种类型的模块：

auth： 这种类型的模块为用户验证提供两方而的服务：让应用程序提示用户输入密码或者其它的标记，确认用八的合法性;通过它的凭证许可权限，设泄组成员关系或者其它优先 权。

account： 这类模块执行棊于非验证的账户管理。它主要用来限制/允许用户对某个服务的访问时间，当前有效的系统资源（最多可以有多少个用户），限制用八的位置（例如：root用户只能从控制台登录）。

session： 这类模块的主要用途是处理为用户提供服务Z前/后盂要做的一些事情，包括:记录打开/关闭数据的信息，监视目录等。

password： 用来升级用户验证标记。

control-flag： 控制标志用来设宜验证成功或者失败后PAM需要作出的反应。因为模块可以层叠，控制标，志可以决定每个模块的重要性。应用程序不会意识到单个模块成功或者失败，它只会收到Linux-PAM库成功或者失败的综合反应信息。

层叠模块的执行顺序取决于/etc/pam.conf文件的入口顺序, 入口列前的模块先执行。 从Linux-PAM 6.0开始可以使用两种语法定义控制标志。简单的一种是使用单一关键词定义控制标志。有四个这样的关键词：required> requisite> sufficient 和 optional。Linux-PAM通过如下方式解释这些关键词：

required： 表示即使某个模块对用户的验证失败，也要等所有的模块都执行完毕之后，PAM才返回错误信息。这样做是为了不让川户知道被哪个模块拒绝。如果对川户验证成功,所有的模块都会返回成功信息。

requisite： 如果特定的模块对用户的验证失败，PAM马上返回一个错误信息，把控制权交回应用程序，不再执行其它模块进行验证。

sufficient： 表示如果-•个用八通过这个模块的验证，PAM结构就立刻返回验证成功信息，把控制权交会丿M丿IJ程序。后面的层替模块即使使川requisite或者required控制标志，也不再执行。如果验证失败sufficient的作用和optional相同。

optional： 表示即使木行指定的模块验证失败，也允许用户享受应用程序捉供的服务。使用这个标志，PAM框架会忽略这这个模块产住的验证错谋，继续顺序执行下•个层叠模 块。

module-path： PAM验证模块的路径。如果以/开头，就表示是完整的路径;如果不是以/打头，就表示是相对于/usr/lib/security的相对路径。

args： 传递给模块的参数。类似于通常的Linux Shell命令行参数。有效的参数包括一些通用参数和特定于给定模块的参数。无效的参数将被忽略，并会把错误信息记录到syslogo

需要特别注总：配置文件中的任何•行错谋都会导致验证失败，同时相关错谋信息被记录到 syslogo

举一个简单的例了如下:

```
? Login auth required pam_unix.so debug
/ Login auth required pam kerb.so use mapped pass                    /
I Login auth optional pam_rsa.so use_first_pass
```

这样，当login程序执行时先用pam_unix.so模块即传统的UNIX 口令方式鉴别用户，然示再调用pam_kerb.so模块即Kerberos对用户进行鉴别，最示用pam_rsa.so模块即RSA方式鉴别用户。在按上述顺序鉴别用八的过程中，如果pam_unix.so模块鉴别失败，它将继续调用下面的模块进行鉴别而非立刻向login程序返回错误消息；

pam_kerb.so模块也按同样方式处理，直到顺序处理完最丿匚-个pam_rsa.so模块后，PAM才将前面出现的错i吴信息返回给login程序。对丁•该配匿，即使pam_rsa.so模块顺利通过，只要pam_unix.so模块和pam_kerb.so模块中有一个出现错课，用户就不能通过鉴别;相反，即使pam_rsa.so模块失败，只要pam_unix.so模块和pam_kerb.so模块都通过了，用户也能通过鉴别。

再举一个有关什卩的例子如卜•:

```
[ftp auth required pam_unix_auth.so debug
```

这样，当用户使用ftp时，将用传统的UNIX 口令鉴别方式来验证其身份。

值得一提的是：在有的Linux操作系统（比如Fedora） I、•并没有现成的/etc/pam.conf文件可以使用，用户需要按照本节的介绍自行进行生成和编辑。

2.2 口令映射机制

在同一个机器上使用多个鉴别机制，尤其是一个应用程序集成多种鉴别机制可能导致川户需要记忆多个口令，这会让用户觉得很不舒服。虽然对以让所有机制使用和同的口令来获收易用性，但是这将削弱系统的安全性■■如果其中任何•个机制的口令泄露了，则所有机制都会受到牵累。

此外，不同的鉴别机制在口令长度、容许的字符、更新间隔、有效期等方面可能具有他们特有的要求，这些要求也是为多鉴别机制使用同一个I I令必须考虑的一个问题。

PAM为我们提供了这样一种解决方案，它不排除为所有鉴别机制共用一个口令，同时允许通过口令映射技术让每个机制使用不同的口令。该方案用用户的“主口令‘加密其他的“副口令“，并且将这些经过加密的副口令存放在一个用户能访问的地方。主口令一旦经过验证，鉴别模块就能用它解密那些加密的副ri令从而获得相应口令，然后将所需口令传递给鉴别模块。这称为”口令映射”。

如果口令映射出现错误，或如果映射不存在，那么各鉴别模块应该提示用户输入口令。为支持口令映射，主口令应保存在PAM第二层并且在需要时由其提供给堆往的各个鉴别模块。同时，口令要在pam_authenticate函数返回Z前清除。为了保障口令映射的安全，主口令必须足够强壮，可以考虑使其的长度更长、组成口令字符的类型多样化并使用混合类型的字

符组成口令等有效措施。

口令如何加密及其存储完全取决丁具体的实现：它能够将加密的副口令他称作”映射口令”）存储在对靠或不町靠的地方，诸如智能R、本地文件或FI录服务。当然，如果加密的口令保存在一个不可靠的允许公共访问的地方，会留下受到字典攻击的隐患。

为实现口令映射，所有鉴别模块应支持以下四个映射选项：

use_first_pass :它表示当该模块执行时不提示用户输入口令，而将该模块之前的提示川户输入的主口令作为它们的公共口令进行验证。如果川户没能通过主口令的鉴别，则该模块不提示用户输入口令。此选项一般说来在系统管理员想强制用同一个口令通过多模块时使 用。

try_first_pass :除了如果主口令不正确，将提示用户输入口令之外，它的用法与use_first_pass 相同。

use_mapped_pass :它表示使用I I令映射技术得到此模块的有效I I令。也就是说，该模块执行时不捉示川户输入口令，而是川映射口令即川主口令解密得到的该模块的副口令作为本模块的口令输入进行验证。如果在此Z前用户没能通过主口令的鉴别，则该模块也不会提示输入口令。

try_mapped_pass :除了如果主口令不正确，它将提示用户输入口令Z外，该项少use_mapped_pass 用法相同。

当口令更换后, PAM会保存所有新旧口令，并且使有关模块能够访问到它们。其他模块能够使用此信息更新加密的口令而不必强制用户再次输入口令。

现以上而使用的有关login的配置文件为例讲解口令映射:

```
? Login auth required pam_unix.so debug                    /
:Login auth required pam kerb.so use_mapped_pass           ?
|                                                          !
? Login auth optional pam rsa.so use_first_pass            :
事                                                         鉴
```

在这里login程序集成了三种鉴别方式：传统UNIX 口令鉴别、Kerberos和RSA鉴别,但通常悄况下用户仅输入一次口令便能通过鉴别了。当程序调用pam_unix.so模块时，PAM提示用八输入他们的UNIX 口令，然后由pam_kerb.so模块对用八输入的UNIX 口令进行 鉴别。

继而调用pam_kerb.so模块，由于该模块的选项为use_mapped_pass,它将利用口令

映射机制进行认证，也就是说，如果UNIX 口令鉴别通过的话，就将具作为pam_kerb.so模块的主口令來解密其对应的映射口令从而进行Kerberos鉴别。

如果pam_unix.so模块所需口令没能通过验证，则无法进行口令映射，那么PAM将直接调用下一鉴别模块而不提示用户输入其Kerberos 口令。最后一个模块的选项为use_first_pass,所以pam_rsa.so模块将使用前边输入的主I I令来鉴别用户，如果I I令错误也不提示用户输入RSA 口令。所以，只要第一次输入的口令是正确的，并映射口令存在，则一个口令便足以通过鉴别。

2.3基于冃录的配置形式

从Linux-PAM 5.6版开始引入了 •种基于口录的配置方式，通过/etc/pam.d/n录下的文件对PAM进行配置。这种方式比单一的配置文件具有更大的灵活性。这个目录下的所有配置文件都以某个服务名命名（小写）。不过，这两种配置方式不能同时起作川，也就是说，自己只能使用其中一种对Linux-PAM进行配置。一般/etc/pam.d/优先。

/etc/pam.d/目录卜的配置文件的语法和/etc/pam.conf文件的语法相似，形式如卜（参见图1所示的关于sshd的PAM配置）：

/ module-type control-flag module-path arguments

和/etc/pam.conf文件语法的唯-不同就是没有服务名（service-name）,服务名由文件名设置。例如：/etc/pam.d/login文件保存对login服务的设置。

这种配置方式与单一配置文件相比，具有很大的优越性：

减少了配置错误的儿率。

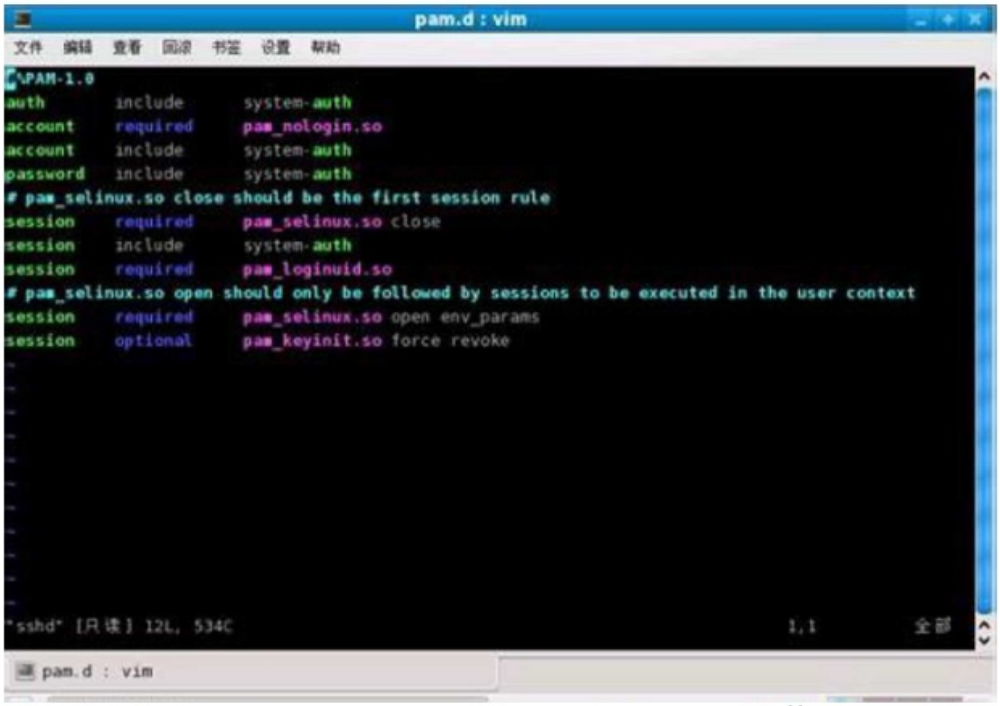
更易于维护。

可以通过使用不同配置文件的符廿连接决定系统的验证策略。

可以加快对于配置文件的解析。

可以对单个的Linux-PAM配置文件设置不同的存取权限。

更易于软件包的管理。



- · pamdE **flWl 簷**

0应用理序＜21 勸I ® Q **三 卩；小 臥5 1阴泪IM5 3卅 硝片**

图1他伽如应配苴文件示意心 i想

3 Linux-PAM使用举例

本节将给出几个使用PAM技术对用户进行安全认证，从而提高系统以及网络服务安全性的例了，來向读者介绍如何在实践中对该机制进行熟练的使川。在系统川户的认证中，我们将介绍如何使用PAM来控制用户縊录;在网络服务安全保证的例子中，我们将介绍如何控制FTP服务器的用户登录问题。

3.1使用Linux-PAM控制用八安全登录

假设系统的安全需求为：控制可以登录系统的用户，只允许root用户可以从本地登录，并且只允许liyang用户可以从192.168.13.8网段远程登录Linux系统，其他用户均不可以 登录系统。

为了完成上述安全控制功能，我们需要修改/etc/pam.d/login文件，加入新的规则如图2中白色区域所示：

❹应
卿覆

图2修改/etc/pam.d/login文件限制登录心

不难看出，我们在原来的基础上加入了以卜•这一条规则（在上述程序段用黑体标明）：

| account required /lib/security/pam_access.so accessfile=/etc/login.conf

这条规则的具体含义是我们使用pam_access模块，通过配置文件/etc/login.conf来对用户访问进行控制，accessfile参数即指明了配宜文件的完整路径。

根据需求，我们的/etc/login.conf文件内容如H:

```
i # vi /etc/login.conf
+:root:LOCAL
  +:liyang:192.168.13.
  ·:ALL:ALL
```

该配置文件的含义如下：

+:root:LOCAL： 表示root用户可以从本地登录

+:liyang:192.168.13.：表示用户 liyang 可以从 192.168.13.8/24 网段远程登录 Linux 系 统

-:ALL:ALL：表示拒绝其他任何人登录

可以清楚地看到，该文件的每一行由如下三个字段构成，中间使用冒号分割，格式为:
权限：用户：来源

权限字段使用“+”表示允许访问，“•”表示禁止访问；

用户字段可以是用户名、组名以及诸如格式的用户名，ALL表示任何人，具有多个值时，可以用空格分开；

来源字段可以是tty名称（木地登录时）、主机名、域名（以开始），主机IP地址，网络号（以“•”结束）。ALL表示任何主机，LOCAL表示本地登录。这里需要特别强调•下：ALL字段在/etc/login.conf文件中出现的顺序非常重要。如果它出现在文件头部，那么后续的其他字段所表示的访问权限将随之失效;如果它出现在其他字段之后，那么其他的字段将牛效。所以•般建议将ALL字段表示的访问权限置于文件末尾处，以保证其他访问权限的正常解 析和使用。

3.2使用Linux-PAM控制Samba用户的共享登录

在Samba中集成PAM的访问控制功能，非常简单，只需要完成如下两个步骤即可。首先，在Samba服务器的配置文件smb.conf的［global］配置段中，保证如下的语句未被屏 蔽：

obey pam restrict!ons = yes

然后，保证有关Samba的PAM配置/etc/pam.d/samba文件正常即可，该配置文件如 图3所示:

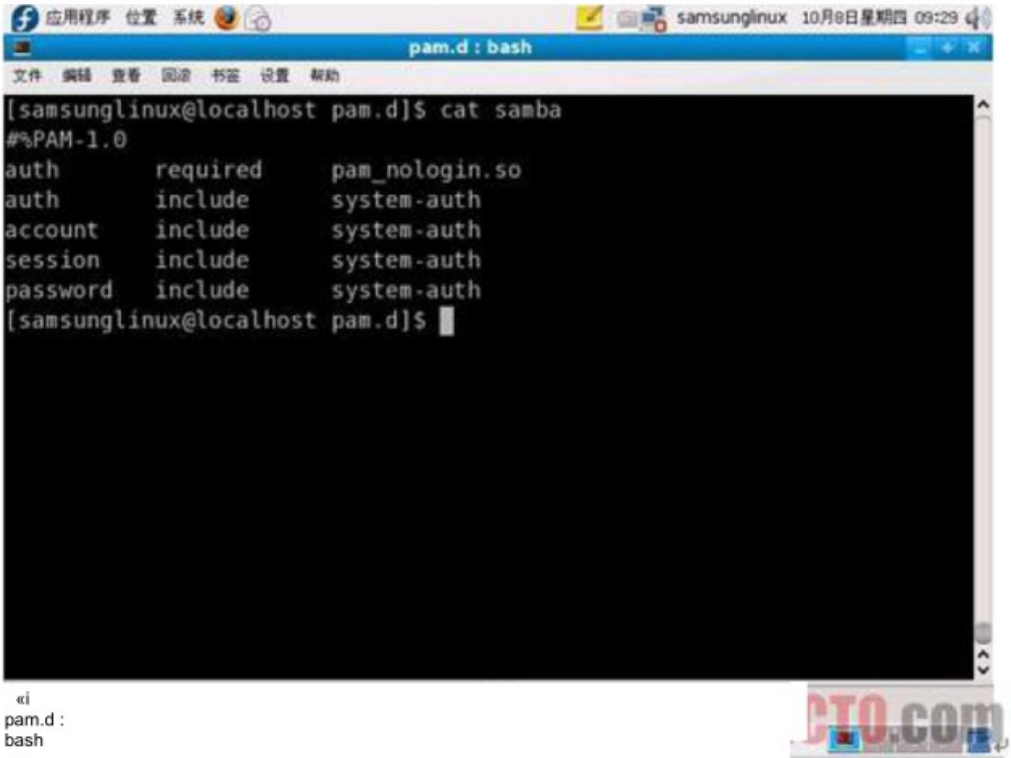


图 3 /etc/pam.d/samba 配置文件示意

3.3使用Linux-PAM控制FTP用户的登录

假设Linux系统提供FTP服务，但是不希望任何用八都可以轻易地访问该服务，只允许在•个事先定义好的文件中规定的用户可以使用该FTP服务。那么，我们就町以使用 Linux-PAM机制来实现该控制。

当然，值得注意的是该FTP服务需要支持Linux-PAMo幸运的是，当前Linux中的proftpd、vsftpd等都对其完全支持。在下而的例子中，我们将以最新的vsftpd为例來进行举例说明。

首先，我们需要定义与认证相关的/etc/pam.d/vsftpd文件如图4所示：

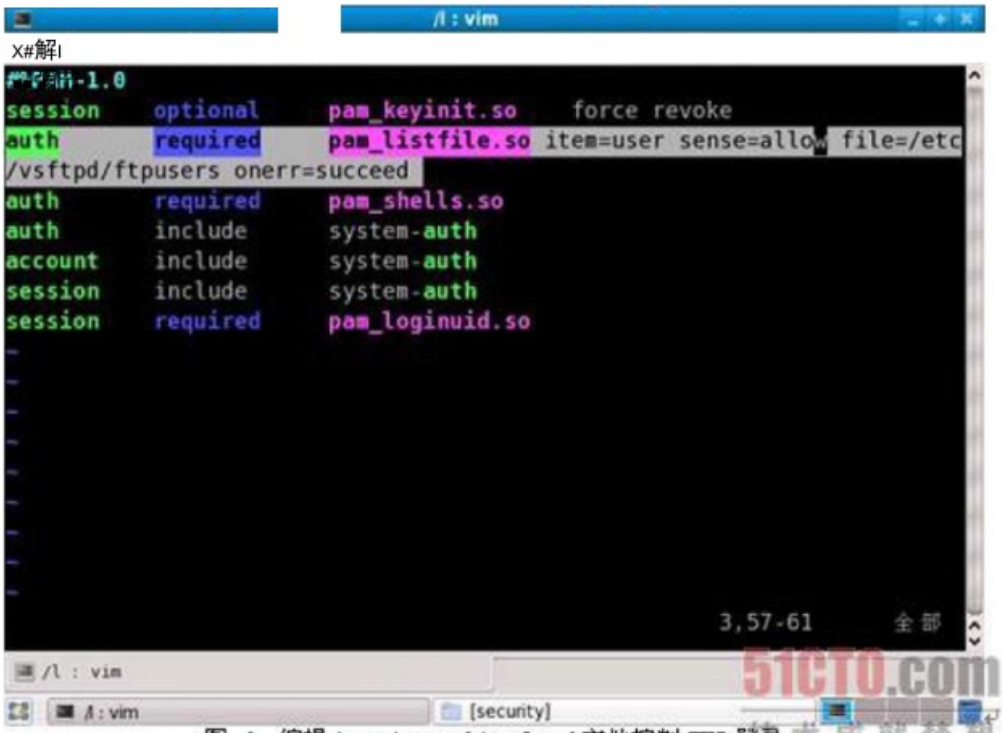


图 4 编辑/etc/pam.d/vsftpd文件控制FTP登录

其中的pam」stfile.so模块就是用来实现基于用户的FTP控制。Item=user就表明是通过用八名进行控制，sense=allow表示如果用八名出现在/etc/vsftpd.ftpusers文件中就返回认证成功信息，file二/etc/vsftpd.ftpusers指定配置文件，onerr=succeed农示如果出现某吐错误（比如无法打开配置文件等）吋返回的结果，这里是成功信息。

然后，我们需要编辑/etc/vsftpd/ftpusers文件，并在其中加入町以进行ftp访问的用户名，耍注意每个用户占一行（如卜-所示），Z后重启动vsftpd即对，就可以根据这个配置文件通过用户名来对FTP访问进行控制了，如图5所示：

Linux授权

类似于Windows系统Linux系统中用户是具有权限属性的，其至它的权限设證更为严格。我们知道在Linux系统中root是管理员用户拥有最高的权限，除此之外的其他用户是普通用八英权限都是受限的。如何让普通用八也具有root权限当管理员使用呢？下面笔者搭建环境，以重启网络操作为例进行Root授权演示。

环境说明：

OS： Fedora Core 6 （域名：ns」inux.com.cn）

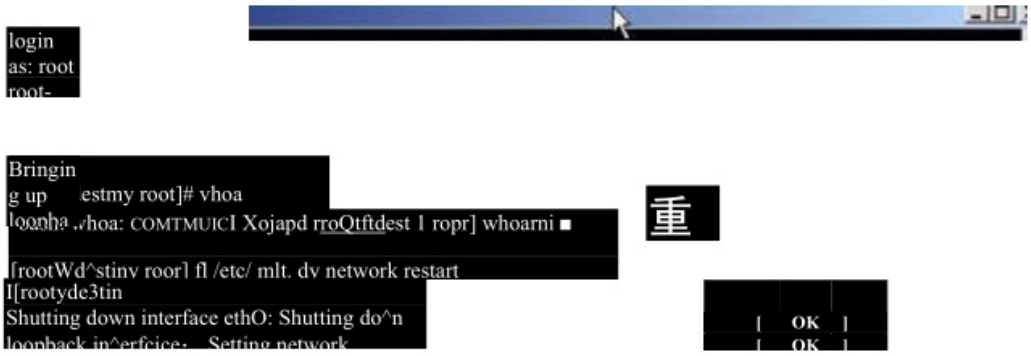
Tools： PuTTY（SSH/Telnet 远程登录）

1＞登录系统

运行PuTTY,首先以Root用户远程登录系统，输入用户名、密码成功背入系统。然后以普通用户hacker远程背录输入用户名、密码进入系统。

2、权限测试

在Root用户登录窗I丨中我们输入命令Vetc/init.d/network restart"重启网络服务，如图7所示启动成功。然后在hacker川户登录窗口输入同样的命令重川网络，显示失败，町以普通用户hacker是没有执行该命令的权限的。（图7）



3、Root授权

要使得普通用/’ hacker也具冇重启网络的权限，我们需要修改/etc/sudoers文件。输入命令“ls -l /etc/sudoers”查看root用户默认对该文件只有“读”权限,是没有办法修改的。对此，我们可以修改其权限止Root用户可以修改，也可以用vi打开该文件修改然后强行保存退出，当然最方便的是用“visudo”命令进行编辑。

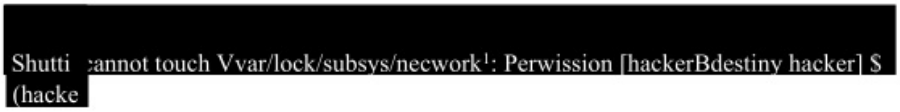
定位至『# User privilege specification” K,扌安照格工T'username host user name command”进行输入。具中第•个字段是用户名（被授权用户），第二个自动是主机位（用域名、IP地址都可以），第三个字段是用户名（授权用户），第四个字段是命令（授权用户可以执行的

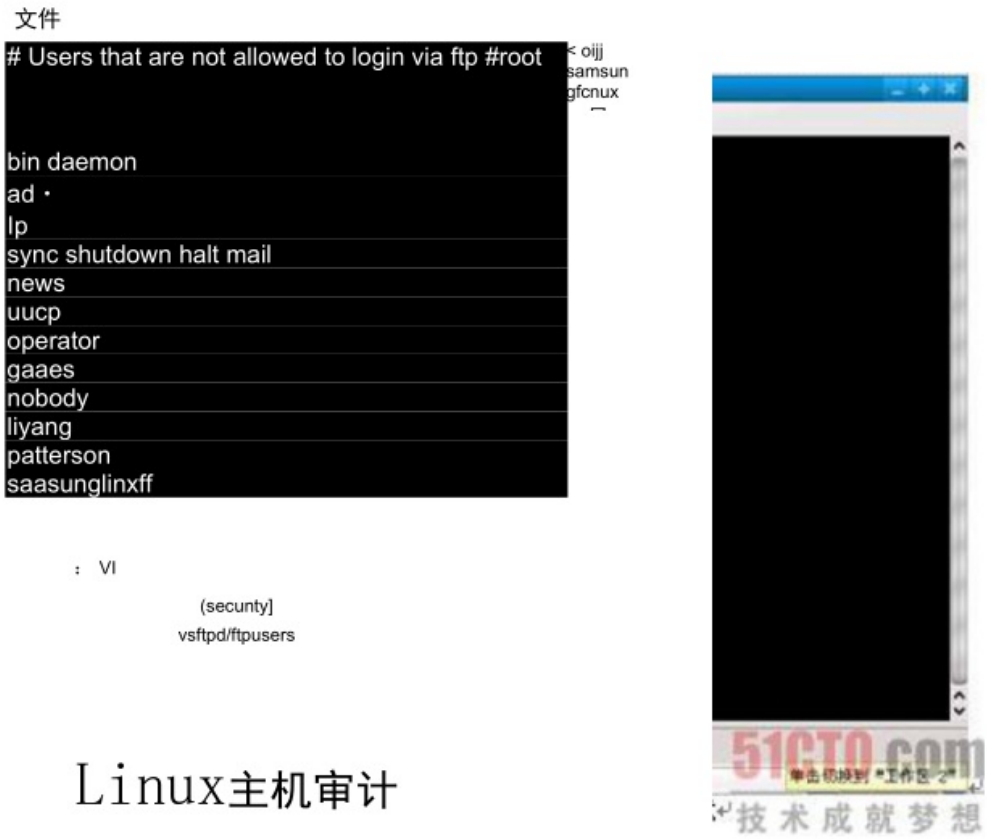
命令，可以用别名）。结合实例我们添加如下字段：hacker All = （root） /etc/init.d/network即授权hacker用户以Root权限运行/etc/init.d/network,最后保存退出。（图8）

4、权限测试

在 hacker 用户窗11中输入命令:sudo /etc/init.d/network restart,看 hacker 是否可以璽启网络，如图9所示命令成功执行网络被重启，说明授权成功。这里必须要说明的是必须以“sudo”来运行命令，因为sudo命令会调用7etc/sudoers[M]脚木，刚才的授权才会生效。

另外，在命令执行前要输入密码，这个密码是当前用户即hacker的密码。（图9）





Linux主机审计

Linux操作系统可以通过设置H志文件可以对每个用户的每一条命令进行纪录，不过这一功能默认是没有打开的。

开启这个功能的过程：

```
# touch /var/log/pacct
# action /var/log/pact
```

也可以用自己的文件来代替/var/log/pacct这个文件。但必须路径和文件名的正确。

sa命令与ac命令一样，sa是一个统计命令。该命令可以获得每个用户或每个命令的进程使用的大致情况，并且提供了系统资源的消费信息。在很大程度上，sa 又是一个记帐命令，对于识别特殊用户，特别是已知特殊用户使用的可疑命令十分有用。另外，由于信息量很大，需要处理脚本或程序筛选这些信息。

lastcomm命令，与sa命令不同，lastcomm命令提供每一个命令的输出结果，同时打印出与执行每个命令有关的时间印戳。就这一点而说，lastcomm比sa更有安全性。如果系统被入侵,请不要相信在lastlog> utmp> wtm中记录的信息，但也不要忽略，因为这些信息可能被修改过了。另外冇可能冇人替换了who程序来掩人耳目。通常，在已经识别某些可疑活动后，进程记帐可以右效的发挥作用。使用lastcomm可以隔绝用户活动或在特定时间执行命令。

3、使用logrorate对审计文件管理

/var/log/utmp,/var/log/wtmp 和/var/log/pacct 文件都是动态的数据文件。wtmp和pacct文件是在文件尾部不断地增加记录。在繁忙的网络上，这些文件会变得很大上inux捉供了一个叫logrotate的程序，它允许管理员对这些文件进行管理。

Log rotate读取/etc/log rotate, d目录下的文件。管理员通过该目录卜•的脚本文件，控制logrotate程序的运作。一个典型的脚本文件如下：

rotate 5

weekly

errors root@serve1r

mail root@server1

copytr un cate

compress

size 100k

}

脚本文件的含义如下：

- rotate 5——保留该文件-•份当前的备份和5份旧的备份。
- weekly——每周处理文件一次，通常是一周的第一天。
- errors——向邮件地址发送错谋报告。
- mail -----向邮件地址发送相关的信息。
- copytruncate-----允许进程持续地记录，备份文件创建后，把活动的H志文件清空。
- compress-----使用gzip工具对旧的日志文件进行压缩。
- size 100k——当文件超过100k时口动处理。