**NSD SECURITY DAY03**

1. [案例1：Linux基本防护措施](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/SECURITY/DAY03/CASE/01/index.html#case1)
2. [案例2：使用sudo分配管理权限](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/SECURITY/DAY03/CASE/01/index.html#case2)
3. [案例3：提高SSH服务安全](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/SECURITY/DAY03/CASE/01/index.html#case3)
4. [案例4：SELinux安全防护](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/SECURITY/DAY03/CASE/01/index.html#case4)

**1 案例1：Linux基本防护措施**

**1.1 问题**

本案例要求练习Linux系统的基本防护措施，完成以下任务：

1. 修改用户zhangsan的账号属性，设置为2019-12-31日失效（禁止登录）
2. 临时锁定用户lisi的账户，使其无法登录，验证效果后解除锁定
3. 修改tty终端提示，使得登录前看到的第一行文本为“Windows Server 2012 Enterprise R2”，第二行文本为“NT 6.2 Hybrid”
4. 锁定文件/etc/resolv.conf、/etc/hosts，以防止其内容被无意中修改
5. 关闭不需要的服务

**1.2 步骤**

实现此案例需要按照如下步骤进行。

步骤一：修改用户zhangsan的账户属性，设置为2019-12-31日失效（禁止登录）

1）正常情况下，未过期的账号可以正常登录，使用chage可以修改账户有效期。

1. chage命令的语法格式：
2. chage -l    账户名称                #查看账户信息
3. chage -E 时间 账户名称                #修改账户有效期

2）失效的用户将无法登录

使用chage命令将用户zhangsan的账户设为当前已失效（比如已经过去的某个时间）：

1. [root@proxy ~]# useradd zhangsan        #创建账户
2. [root@proxy ~]# passwd zhangsan        #设置密码
3. [root@proxy ~]# chage -E 2017-12-31 zhangsan    #设置账户过期时间

尝试以用户zhangsan重新登录，输入正确的用户名、密码后直接闪退，返回登录页，说明此帐号已失效。

3）重设用户zhangsan的属性，将失效时间设为2019-12-31

1. [root@proxy ~]# chage -E 2019-12-31 zhangsan             #修改失效日期
2. [root@proxy ~]# chage -l zhangsan                     #查看账户年龄信息
3. Last password change                     : May 15, 2017
4. Password expires                     : never
5. Password inactive                     : never
6. Account expires                         : Dec 31, 2019
7. Minimum number of days between password change         : 0
8. Maximum number of days between password change         : 99999
9. Number of days of warning before password expires         : 7

4）重设用户zhangsan的属性，将失效时间设为永不过期

1. [root@proxy ~]# chage -E -1 zhangsan #设置账户永不过期

5）定义默认有效期（扩展知识）

/etc/login.defs这个配置文件，决定了账户密码的默认有效期。

1. [root@proxy ~]# cat /etc/login.defs
2. PASS\_MAX\_DAYS    99999                        #密码最长有效期
3. PASS\_MIN\_DAYS    0                            #密码最短有效期
4. PASS\_MIN\_LEN    5                            #密码最短长度
5. PASS\_WARN\_AGE    7                            #密码过期前几天提示警告信息
6. UID\_MIN 1000                #UID最小值
7. UID\_MAX 60000                #UID最大值

步骤二：临时锁定用户zhangsan的账户，使其无法登录，验证效果后解除锁定

1）锁定用户账号

使用passwd或usermod命令将用户zhangsan的账户锁定。

1. [root@proxy ~]# passwd -l zhangsan         #锁定用户账号（lock）
2. 锁定用户 zhangsan 的密码。
3. passwd: 操作成功
4. [root@proxy ~]# passwd -S zhangsan         #查看状态（status）
5. zhangsan LK 2018-02-22 0 99999 7 -1 (密码已被锁定。)

2）验证用户zhangsan已无法登录，说明锁定生效

输入正确的用户名、密码，始终提示“Login incorrect”，无法登录。

3）解除对用户zhangsan的锁定

1. [root@proxy ~]# passwd -u zhangsan         #解锁用户账号（unlock）
2. 解锁用户 zhangsan 的密码 。
3. passwd: 操作成功
4. [root@proxy ~]# passwd -S zhangsan         #查看状态
5. zhangsan PS 2018-08-14 0 99999 7 -1 (密码已设置，使用 SHA512 加密。)

步骤三：修改tty登录的提示信息，隐藏系统版本

1）账户在登录Linux系统时，默认会显示登陆信息（包括操作系统内核信息）

/etc/issue这个配置文件里保存的就是这些登陆信息，修改该文件防止内核信息泄露。

1. [root@proxy ~]# cat /etc/issue                     #确认原始文件
2. Red Hat Enterprise Linux Server release 7.5
3. Kernel \r on an \m
4. [root@proxy ~]# cp /etc/issue /etc/issue.origin     #备份文件
5. [root@proxy ~]# vim /etc/issue                     #修改文件内容
6. Windows Server 2012 Enterprise R2
7. NT 6.2 Hybrid

2）测试版本伪装效果

退出已登录的tty终端，或者重启Linux系统，刷新后的终端提示信息会变成自定义的文本内容，如图-1所示。



图-1

步骤四：锁定文件/etc/resolv.conf、/etc/hosts

1）语法格式：

1. # chattr +i 文件名            #锁定文件（无法修改、删除、追加等）
2. # chattr -i 文件名            #解锁文件
3. # chattr +a 文件名            #锁定后文件仅可追加
4. # chattr -a 文件名            #解锁文件
5. # lsattr 文件名                #查看文件特殊属性

2) 使用+i锁定文件，使用lsattr查看属性

1. [root@proxy ~]# chattr +i /etc/resolv.conf
2. [root@proxy ~]# lsattr /etc/resolv.conf
3. ----i---------- /etc/resolv.conf

3）使用+a锁定文件(仅可追加)，使用lsattr查看属性

1. [root@proxy ~]# chattr +a /etc/hosts
2. [root@proxy ~]# lsattr /etc/hosts
3. -----a---------- /etc/hosts

4）测试文件锁定效果

1. [root@proxy ~]# rm -rf /etc/resolv.conf
2. rm: 无法删除"/etc/resolv.conf": 不允许的操作
3. [root@proxy ~]# echo xyz > /etc/resolv.conf
4. -bash: resolv.conf: 权限不够
5. [root@proxy ~]# rm -rf /etc/hosts                      #失败
6. [root@proxy ~]# echo "192.168.4.1 xyz" > /etc/hosts #失败
7. [root@proxy ~]# echo "192.168.4.1 xyz" >> /etc/hosts #成功

5）恢复这两个文件原有的属性（避免对后续实验造成影响）

1. [root@proxy ~]# chattr -i /etc/resolv.conf
2. [root@proxy ~]# chattr -i /etc/hosts
3. [root@proxy ~]# lsattr /etc/resolv.conf /etc/hosts
4. --------------- /etc/resolv.conf
5. --------------- /etc/hosts

步骤五：关闭不需要的服务

使用systemctl、chkconfig工具，禁用非必要的系统服务。

可选服务列表：

cups.service 打印服务

postfix.service 邮件服务

NetworkManager.service 网络管理服务(network可以替代）

firewalld 防火墙（iptables可以替代）

atd.service 一次性计划任务（crond可以替代）

bluetooth.service 蓝牙服务

autofs.service 自动挂载

pcscd.service 智能卡设备资源管理器

**2 案例2：使用sudo分配管理权限**

**2.1 问题**

本案例要求利用sudo机制分配管理操作权限，主要完成以下任务：

1. 使用su命令临时切换账户身份，并执行命令
2. 允许softadm管理系统服务的权限
3. 允许用户useradm通过sudo方式添加/删除/修改除root以外的用户账号
4. 允许wheel组成员以特权执行所有命令
5. 为sudo机制启用日志记录，以便跟踪sudo执行操作

**2.2 步骤**

实现此案例需要按照如下步骤进行。

步骤一：使用su命令临时切换账户身份

su(Substitute User)命令可以快速切换账户身份，普通用户切换账户身份时需要输入密码，root使用su命令切换任何身份都不需要密码，如法格式如下：

1. # su - [账户名称]
2. # su - [账户名称] -c '命令'

1)从普通用户切换为root账户身份(如果没有普通账户则需要先创建，并设置密码)

1. [zhangsan@proxy ~]# whoami
2. zhangsan
3. [zhangsan@proxy ~]# su -            #切换账户，默认切换为root账户
4. 密码: #输入root的密码
5. [root@proxy ~]# whoami                #确认结果
6. root

2)以普通身份创建文件(如果没有普通账户则需要先创建)

1. [root@proxy ~]# su - zhangsan -c "touch /tmp/test.txt"        #管理员切换普通用户
2. [root@proxy ~]# ls -l /tmp/test.txt                        #查看文件详细信息

3)以root的身份重启服务

1. [zhangsan@proxy ~]# su - -c "systemctl restart sshd"     #普通用户切换管理员
2. 密码：
3. ● sshd.service - OpenSSH server daemon
4. Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
5. active: active (running) since 五 2018-01-19 08:59:40 CST; 1 months 4 days ago

步骤二：允许softadm管理系统服务的权限

1）修改/etc/sudoers配置

修改/etc/sudoers可以直接使用vim编辑该文件，或使用visudo命令修改该文件。

授权的语法格式：用户或组 主机名=（提权账户） 命令

注释：授权什么人，在什么主机上，以什么人的身份，执行什么命令。

为softadm授予通过systemctl工具来管理系统服务的权限。

如果没有softadm账户可以先创建该账户。

1. [root@proxy ~]# useradd softadm            #创建账户
2. [root@proxy ~]# passwd softadm                #设置密码
3. [root@proxy ~]# vim /etc/sudoers            #修改文件后，需要使用wq!强制保存
4. .. ..
5. softadm ALL=(ALL) /usr/bin/systemctl
6. #授权softadm以root身份执行systemctl命令（ALL包括root）

2）切换为softadm用户，并验证sudo执行权限

1. [root@proxy ~]# su - softadm
2. [softadm@proxy ~]$ sudo -l
3. … …
4. [sudo] password for softadm:                 #输入softadm的口令
5. .. ..
6. 用户 softadm 可以在该主机上运行以下命令：
7. (ALL) /usr/bin/systemctl
8. [softadm@proxy ~]$ systemctl start crond #不用sudo时启动服务失败
9. Authentication is required
10. .. ..
11. [softadm@proxy ~]$ sudo systemctl restart crond         #通过sudo启动服务成功

步骤三：允许用户useradm通过sudo方式添加/删除/修改除root以外的用户账号

1）修改/etc/sudoers配置

为useradm授予用户管理相关命令的执行权限，例外程序以!符号取反，放在后面。在执行相关程序时，可以利用通配符\*。

1. [root@proxy ~]# useradd useradm        #创建账户
2. [root@proxy ~]# passwd useradm        #设置密码
3. [root@proxy ~]# vim /etc/sudoers        #修改文件，需要:wq!强制保存退出
4. .. ..
5. useradm ALL=(ALL) /usr/bin/passwd,!/usr/bin/passwd root,/usr/sbin/user\*,!/usr/sbin/user\* \* root

2）切换为useradm用户，验证sudo权限

可以通过sudo方式来添加/删除/修改普通用户：

1. [useradm@proxy ~]$ sudo -l
2. .. ..
3. 用户useradm可以在该主机上运行以下命令：
4. (root) /usr/bin/passwd, !/usr/bin/passwd root, /usr/sbin/user\*,!/usr/sbin/user\* \* root
5. [useradm@proxy ~]$ sudo useradd newuser01                 #可以添加用户
6. [useradm@proxy ~]$ sudo passwd newuser01                 #可以修改普通用户的口令
7. 更改用户 newuser01 的密码 。
8. 新的 密码：
9. 重新输入新的 密码：
10. passwd： 所有的身份验证令牌已经成功更新。

但是不能修改root用户的密码：

1. [useradm@proxy ~]$ sudo passwd root
2. 对不起，用户 useradm 无权以 root 的身份在 localhost 上
3. 执行 /usr/bin/passwd root。

步骤四：允许wheel组成员以特权执行所有命令

此案例用来展示sudo的便利性及设置不当带来的危险性，生产环境下慎用。

实现时参考下列操作(如果没有普通用户zengye则先创建该账户)：

1. [root@proxy ~]# useradd zengye        #创建账户
2. [root@proxy ~]# passwd zengye        #设置密码
3. [root@proxy ~]# cat /etc/sudoers    #查看文件即可
4. .. ..
5. %wheel ALL=(ALL) ALL
6. [root@proxy ~]# usermod -G wheel zengye    #设置zengye用户的附加组为wheel
7. [root@proxy ~]# su - zengye
8. [zengye@proxy ~]$ sudo -l
9. .. ..
10. 用户 zengye 可以在该主机上运行以下命令：
11. (root) /bin/\*

步骤五：为sudo机制启用日志记录，以便跟踪sudo执行操作

1）修改/etc/sudoers配置，添加日志设置（使用visudo也可以修改配置文件）

1. [root@proxy ~]# visudo
2. Defaults logfile="/var/log/sudo"
3. .. ..

2）执行sudo操作产生日志

使用root可以执行sudo命令，softadm也可以执行sudo命令（前面已经授权）。

1. [root@proxy ~]# sudo -l                 #查看授权的sudo操作
2. [softadm@proxy ~]# sudo systemctl status crond             #执行sudo重启服务

3）确认日志记录已生效

1. [root@proxy ~]# cat /var/log/sudo         #查看日志文件
2. .. ..
3. May 16 22:14:49 : root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=list
4. Feb 22 22:35:43 : softadm : TTY=pts/11 ; PWD=/home/softadm ; USER=root ;
5. COMMAND=/bin/systemctl status httpd

**3 案例3：提高SSH服务安全**

**3.1 问题**

本案例要求提高Linux主机上SSH服务端的安全性，完成以下任务：

1. 配置基本安全策略（禁止root）
2. 针对SSH访问采用仅允许的策略，未明确列出的用户一概拒绝登录
3. 实现密钥验证登录（私钥口令）、免密码登入
4. 确认密钥验证使用正常后，禁用口令验证

**3.2 步骤**

实现此案例需要按照如下步骤进行。

步骤一：配置基本安全策略

1）调整sshd服务配置，并重载服务

1. [root@proxy ~]# vim /etc/ssh/sshd\_config
2. .. ..
3. PermitRootLogin no                     #禁止root用户登录
4. UseDNS no                         #不解析客户机地址
5. LoginGraceTime 1m                     #登录限时
6. MaxAuthTries 3                     #每连接最多认证次数
7. .. ..
8. [root@proxy ~]# systemctl restart sshd

2）测试基本安全策略

尝试以root用户SSH登录，失败：

1. [root@proxy ~]# ssh root@192.168.4.5
2. root@192.168.4.5's password:
3. Permission denied, please try again.

步骤二：针对SSH访问采用仅允许的策略，未明确列出的用户一概拒绝登录

1）调整sshd服务配置，添加AllowUsers策略，仅允许用户zhangsan、tom、useradm，其中useradm只能从网段192.168.2.0/24远程登录本机。

注意：如果没有这些用户，需要提前创建用户并设置密码。

1. [root@proxy ~]# vim /etc/ssh/sshd\_config
2. .. ..
3. AllowUsers zhangsan tom useradm@192.168.2.0/24    #定义账户白名单（根据实际情况填写）
4. ##DenyUsers USER1 USER2                    #定义账户黑名单
5. ##DenyGroups GROUP1 GROUP2                    #定义组黑名单
6. ##AllowGroups GROUP1 GROUP2                    #定义组白名单
7. [root@proxy ~]# systemctl restart sshd

2）验证SSH访问控制，未授权的用户将拒绝登录。

1. [root@proxy ~]# ssh useradm@192.168.2.5             #已授权的用户允许登录
2. useradm@192.168.2.5's password:
3. [useradm@proxy ~]$ exit
4. [root@proxy ~]# ssh root@192.168.2.5                 #未授权的用户被拒绝登录
5. root@192.168.2.5's password:
6. Permission denied, please try again.

步骤三：实现密钥对验证登录（私钥口令）、免密码登入

1）准备客户机测试环境

为客户机的用户root建立SSH密钥对

使用ssh-keygen创建密钥对，将私钥口令设为空（直接回车）：

1. [root@client ~]$ ssh-keygen
2. Generating public/private rsa key pair.
3. Enter file in which to save the key (/root/.ssh/id\_rsa):
4. Created directory '/root/.ssh'.
5. Enter passphrase (empty for no passphrase):         #直接回车将口令设为空
6. Enter same passphrase again:                         #再次回车确认
7. [root@client ~]$ ls -lh ~/.ssh/id\_rsa\*             #确认密钥对文件
8. -rw-------. 1 root root 1.8K 8月 15 10:35 /root/.ssh/id\_rsa
9. -rw-r--r--. 1 root root 403 8月 15 10:35 /root/.ssh/id\_rsa.pub

2）将客户机上用户root的公钥部署到SSH服务器

以用户root登入客户机，使用ssh-copy-id命令将自己的公钥部署到服务器：

1. [root@client ~]$ ssh-copy-id root@192.168.4.5
2. root@192.168.4.5's password:
3. Now try logging into the machine, with "ssh 'root@192.168.4.5'", and check in:
4. .ssh/authorized\_keys
5. to make sure we haven't added extra keys that you weren't expecting.

3）在服务器上确认客户机用户root上传的公钥信息

默认部署位置为目标用户的家目录下 ~/.ssh/authorized\_keys文件：

1. [root@proxy ~]# cat ~/.ssh/authorized\_keys
2. ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAzz+5AiFMGQ7LfuiV7eBnOcmRO9JRTcqRoynGO2y5
3. RyFL+LxR1IpEbkNrUyIZDk5uaX1Y8rwsf+pa7UZ2NyqmUEvNSUo0hQyDGsU9SPyAdzRCCvDgwpOFhaHi/OFnT+zqjAqXH2M9fFYEVUU4PIVL8HT19zCQRVZ/q3acQA34UsQUR0PpLJAobsf1BLe2EDM8BsSHckDGsNoDT9vk+u3e83RaehBMuy1cVEN5sLAaIrIeyM8Q0WxQNlqknL908HRkTlTeKrRoHbMnOBFj8StwlnscKHlkrsKkhUf8A9WWz/vL4GDwGND5jdca3I2hdITAySjMdfL1HMHnMYOgMjPM0Q== root@192.168.4.100

4）在客户机上测试SSH密钥对验证

在客户机用户root的环境中，以远程用户root登入192.168.4.5主机时，无需验证口令即可登入（因为私钥口令为空）：

1. [root@client ~]$ ssh root@192.168.4.5                     #免交互直接登入
2. Last login: Thu Aug 15 10:48:09 2013 from 192.168.4.100

步骤四：确认密钥验证使用正常后，禁用口令验证

1）调整sshd服务配置，将PasswordAuthentication设为no

1. [root@proxy ~]# vim /etc/ssh/sshd\_config
2. .. ..
3. PasswordAuthentication no                             #将此行yes改成no
4. [root@proxy ~]# systemctl restart sshd

**4 案例4：SELinux安全防护**

**4.1 问题**

本案例要求熟悉SELinux防护机制的开关及策略配置，完成以下任务：

1. 将Linux服务器的SELinux设为enforcing强制模式
2. 从/root目录下移动一个包文件到FTP下载目录，调整策略使其能够被下载

**4.2 步骤**

实现此案例需要按照如下步骤进行。

步骤一：将Linux服务器的SELinux设为enforcing强制模式

1）固定配置：修改/etc/selinux/config文件

确认或修改SELINUX为enforcing模式：

1. [root@proxy ~]# vim /etc/selinux/config
2. SELINUX=enforcing                                #设置SELinux为强制模式
3. SELINUXTYPE=targeted                            #保护策略为保护主要的网络服务安全

2）临时配置：使用setenforce命令

查看当前SELinux状态，如果是disabled则需要根据第1）步的配置重启系统；如果是permissive则使用setenforce命令修改为enforcing即可：

1. [root@proxy ~]# getenforce                        #查看当前状态为警告模式
2. Permissive
3. [root@proxy ~]# setenforce 1                     #设置SELinux为强制模式
4. [root@proxy ~]# getenforce                        #查看当前模式为强制模式
5. Enforcing
6. [root@proxy ~]# setenforce 0                     #设置SELinux为宽松模式
7. [root@proxy ~]# getenforce                        #查看当前模式为宽松模式
8. Permissive

步骤二：在SELinux启用状态下，调整策略打开vsftpd服务的匿名上传访问

1）配置一个允许匿名上传的vsftpd服务作为测试环境

1. [root@proxy ~]# setenforce 1
2. [root@proxy ~]# yum -y install vsftpd
3. [root@proxy ~]# systemctl start vsftpd                #启动服务
4. #默认Vsftpd共享目录为/var/ftp/

步骤三：从/root目录下移动2个包文件到FTP下载目录，调整文件的安全上下文

1）建立两个FTP下载用的测试文件

由root用户创建两个测试压缩包，一个直接建立到/var/ftp/目录下，另一个先在/root/下建立，然后移动至/var/ftp/目录。

1. #测试文件1，直接在ftp目录下创建文件
2. [root@proxy ~]# tar -czf /var/ftp/log1.tar /var/log
3. [root@proxy ~]# ls -lh /var/ftp/
4. -rw-r--r--. 1 root root 8M 8月 16 10:16 log1.tar
5. [root@proxy ~]# ls -Z /var/ftp/
6. -rw-r--r--. root root unconfined\_u:object\_r:public\_content\_t:s0 log1.tar
7. #测试文件2，在/root下建立，然后移动至/var/ftp目录
8. [root@proxy ~]# tar -czf log2.tar /var/log
9. [root@proxy ~]# mv log2.tar /var/ftp/
10. [root@proxy ~]# ls -lh /var/ftp/
11. -rw-r--r--. 1 root root 8M 8月 16 10:16 log2.tar
12. [root@proxy ~]# ls -Z /var/ftp/
13. -rw-r--r--. 1 root root unconfined\_u:object\_r:admin\_home\_t:s0 log2.tar

3）通过FTP方式测试下载

使用wget命令分别下载这两个包文件，第二个包将会下载失败（看不到文件）。

1. [root@proxy ~]# wget ftp://192.168.4.5/log1.tar            #下载第一个文件，成功
2. [root@proxy ~]# wget ftp://192.168.4.5/log2.tar            #下载第二个文件，失败

4）检查该测试包的安全上下文，正确调整后再次下载第二个包成功。

文件已经存放到共享目录下，但客户端无法访问下载，是因为被SELinux拦截了！

1. [root@proxy ~]# ls -Z /var/ftp/
2. -rw-r--r--. root root unconfined\_u:object\_r:public\_content\_t:s0 log1.tar
3. -rw-r--r--. 1 root root unconfined\_u:object\_r:admin\_home\_t:s0 log2.tar
4. [root@proxy ~]# chcon -t public\_content\_t /var/ftp/log2.tar
5. [root@proxy ~]# ls -Z /var/ftp/log2.tar
6. -rw-r--r--. root root unconfined\_u:object\_r:public\_content\_t:s0 log2.tar
7. [root@proxy ~]# wget ftp://192.168.4.5/log2.tar            #再次下载，成功

注意：上例中的chcon操作可替换为（效果相同）：

# restorecon /var/ftp/log2.tar

附加思维导图，如图-2所示：

