

《信息安全综合实践》实验报告

实验名称： 操作系统安全

姓名： 黎锦灏 学号： 518021910771 邮箱： ljh2000@sjtu.edu.cn 实验时长： 75 分钟

一、实验目的

- 1. 了解 Linux 系统下的基础操作，熟悉 Linux 文件管理的基本概念；
- 2. 了解 Windows 操作系统的账户策略管理；
- 3. 了解 Windows 操作系统账户口令、文件系统方面安全设置。

二、实验内容

序	内容	实验内容
1)	Linux 基本命令	(选做)
2)	Linux 文件管理	文件权限查看、修改与添加
3)	Windows 账户策略管理	文件系统安全设置
4)		EFS 加密硬盘数据
5)	Windows 安全策略与审计	安全策略的设置与审计

三、实验过程截图（30 分）

注：将下列截图保留，并用简短的话描述实验所得的结果。

- 1. 实验 2 步骤 1 中将所有文件修改所有者为 hongya，并恢复所有者为 root；
 - ①、首先使用 useradd 创建一个名为 hongya 的普通用户，然后修改 bin 目录下所有文件的所有者为 hongya，再使用 ls -l 命令查看修改结果，可以观察到文件所有者（第三列数据）已经修改为 hongya 了，如图 1 所示。

☆ Linux文件系统管理

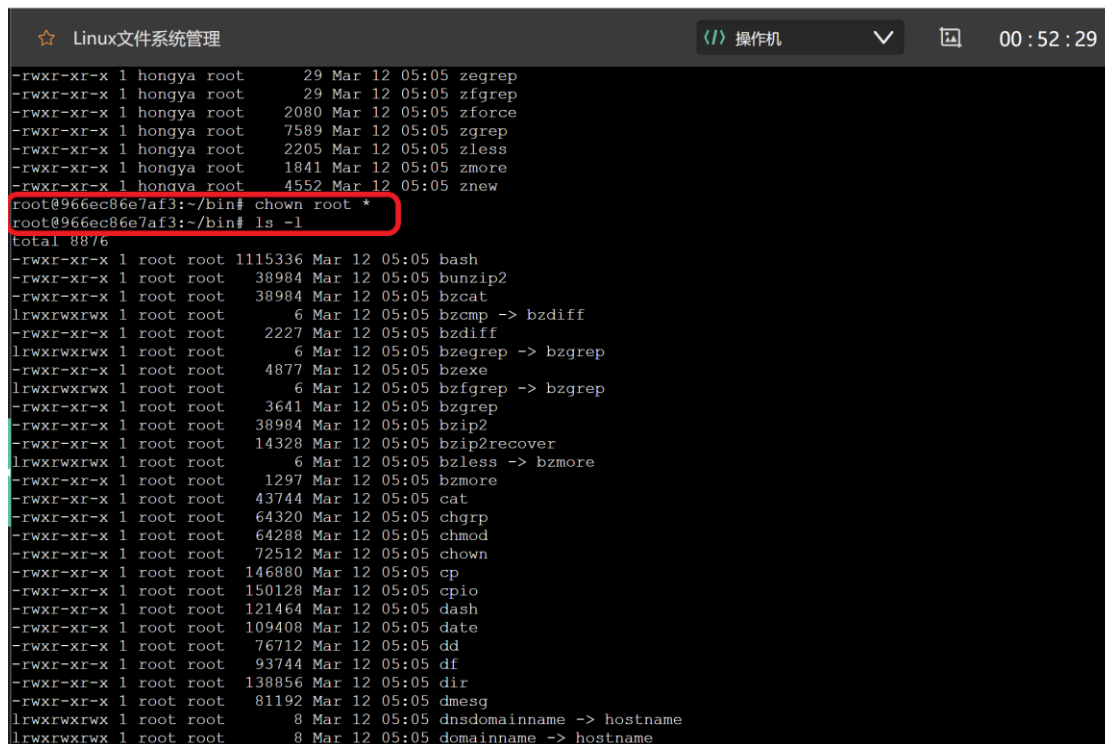
操作机

00:54:18

```
-rwxr-xr-x 1 root root 26704 Mar 12 05:05 kill
-rwxr-xr-x 1 root root 157880 Mar 12 05:05 kmod
-rwxr-xr-x 1 root root 166664 Mar 12 05:05 less
-rwxr-xr-x 1 root root 10256 Mar 12 05:05 lessecho
lrwxrwxrwx 1 root root 8 Mar 12 05:05 lessfile -> lesspipe
-rwxr-xr-x 1 root root 10256 Mar 12 05:05 lesskey
root@966ec86e7af3:~/bin# useradd -g root hongya
root@966ec86e7af3:~/bin# chown hongya *
root@966ec86e7af3:~/bin# ls -l
total 8876
-rwxr-xr-x 1 hongya root 1115336 Mar 12 05:05 bash
-rwxr-xr-x 1 hongya root 38984 Mar 12 05:05 bunzip2
-rwxr-xr-x 1 hongya root 38984 Mar 12 05:05 bzcac
lrwxrwxrwx 1 root root 6 Mar 12 05:05 bzcmp -> bzdiff
-rwxr-xr-x 1 hongya root 2227 Mar 12 05:05 bzdiff
lrwxrwxrwx 1 root root 6 Mar 12 05:05 bzegrep -> bzgrep
-rwxr-xr-x 1 hongya root 4877 Mar 12 05:05 bzexe
lrwxrwxrwx 1 root root 6 Mar 12 05:05 bzfgrep -> bzgrep
-rwxr-xr-x 1 hongya root 3641 Mar 12 05:05 bzgrep
-rwxr-xr-x 1 hongya root 38984 Mar 12 05:05 bzip2
-rwxr-xr-x 1 hongya root 14328 Mar 12 05:05 bzip2recover
lrwxrwxrwx 1 root root 6 Mar 12 05:05 bzless -> bzmoe
-rwxr-xr-x 1 hongya root 1297 Mar 12 05:05 bzmoe
-rwxr-xr-x 1 hongya root 43744 Mar 12 05:05 cat
-rwxr-xr-x 1 hongya root 64320 Mar 12 05:05 chgrp
-rwxr-xr-x 1 hongya root 64288 Mar 12 05:05 chmod
-rwxr-xr-x 1 hongya root 72512 Mar 12 05:05 chown
-rwxr-xr-x 1 hongya root 146880 Mar 12 05:05 cp
-rwxr-xr-x 1 hongya root 150128 Mar 12 05:05 cpio
-rwxr-xr-x 1 hongya root 121464 Mar 12 05:05 dash
-rwxr-xr-x 1 hongya root 109408 Mar 12 05:05 date
-rwxr-xr-x 1 hongya root 76712 Mar 12 05:05 dd
-rwxr-xr-x 1 hongya root 93744 Mar 12 05:05 df
-rwxr-xr-x 1 hongya root 138856 Mar 12 05:05 dir
-rwxr-xr-x 1 hongya root 81192 Mar 12 05:05 dmesg
```

图 1. 修改文件所有者

②、然后用命令 `chown root *` 将文件所有者恢复为 root，用 `ls -l` 可以显示出当前所有者状态。可以看到所有者已经全部是 root 了，如图 2 所示。

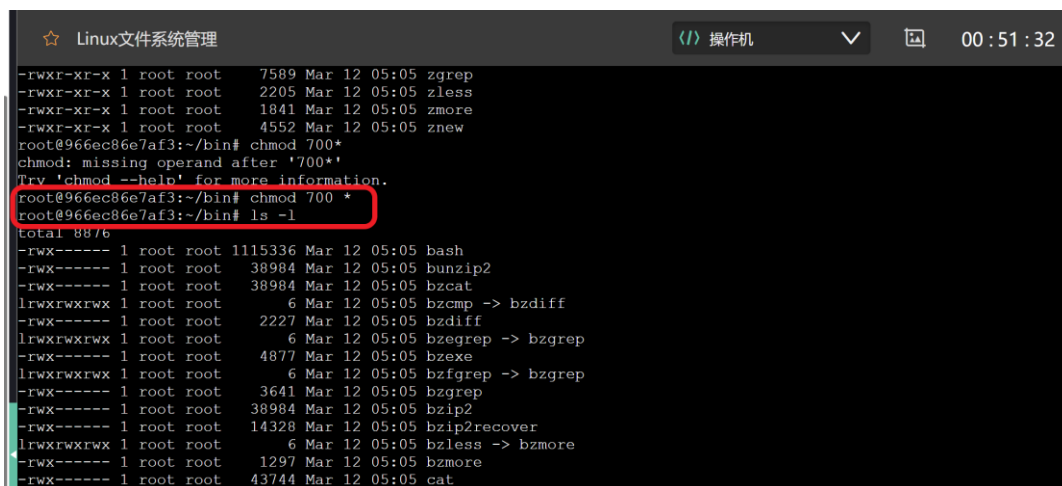


```
Linux文件系统管理 操作机 00:52:29
-rwxr-xr-x 1 hongya root    29 Mar 12 05:05 zegrep
-rwxr-xr-x 1 hongya root    29 Mar 12 05:05 zfgrep
-rwxr-xr-x 1 hongya root   2080 Mar 12 05:05 zforce
-rwxr-xr-x 1 hongya root   7589 Mar 12 05:05 zgrep
-rwxr-xr-x 1 hongya root   2205 Mar 12 05:05 zless
-rwxr-xr-x 1 hongya root   1841 Mar 12 05:05 zmore
-rwxr-xr-x 1 hongya root   4552 Mar 12 05:05 znew
root@966ec86e7af3:~/bin# chown root *
root@966ec86e7af3:~/bin# ls -l
total 8876
-rwxr-xr-x 1 root root 1115336 Mar 12 05:05 bash
-rwxr-xr-x 1 root root  38984 Mar 12 05:05 bunzip2
-rwxr-xr-x 1 root root  38984 Mar 12 05:05 bzcata
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzcmp -> bzdiff
-rwxr-xr-x 1 root root  2227 Mar 12 05:05 bzdiff
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzegrep -> bzgrep
-rwxr-xr-x 1 root root  4877 Mar 12 05:05 bzexe
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzfgrep -> bzgrep
-rwxr-xr-x 1 root root  3641 Mar 12 05:05 bzgrep
-rwxr-xr-x 1 root root  38984 Mar 12 05:05 bzip2
-rwxr-xr-x 1 root root 14328 Mar 12 05:05 bzip2recover
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzless -> bzmore
-rwxr-xr-x 1 root root  1297 Mar 12 05:05 bzmore
-rwxr-xr-x 1 root root 43744 Mar 12 05:05 cat
-rwxr-xr-x 1 root root 64320 Mar 12 05:05 chgrp
-rwxr-xr-x 1 root root 64288 Mar 12 05:05 chmod
-rwxr-xr-x 1 root root  72512 Mar 12 05:05 chown
-rwxr-xr-x 1 root root 146880 Mar 12 05:05 cp
-rwxr-xr-x 1 root root 150128 Mar 12 05:05 cpio
-rwxr-xr-x 1 root root 121464 Mar 12 05:05 dash
-rwxr-xr-x 1 root root 109408 Mar 12 05:05 date
-rwxr-xr-x 1 root root  76712 Mar 12 05:05 dd
-rwxr-xr-x 1 root root  93744 Mar 12 05:05 df
-rwxr-xr-x 1 root root 138856 Mar 12 05:05 dir
-rwxr-xr-x 1 root root  81192 Mar 12 05:05 dmesg
lrwxrwxrwx 1 root root    8 Mar 12 05:05 dnsdomainname -> hostname
lrwxrwxrwx 1 root root    8 Mar 12 05:05 domainname -> hostname
```

图 2. 将文件所有者恢复为 root

2. 实验 2 步骤 2 中修改文件权限结果：

输入 `chmod 700 *` 命令，修改主目录下的文件权限，700 代表 `-rwx-----`，表示文件所有者可读、写、运行，而其他不可读、写、运行的模式。从图 3 中可以看出，主目录下文件权限已经由原来的 `-rwxr-xr-x` 修改为期望的 `-rwx-----`。



```
Linux文件系统管理 操作机 00:51:32
-rwxr-xr-x 1 root root    7589 Mar 12 05:05 zegrep
-rwxr-xr-x 1 root root   2205 Mar 12 05:05 zless
-rwxr-xr-x 1 root root   1841 Mar 12 05:05 zmore
-rwxr-xr-x 1 root root   4552 Mar 12 05:05 znew
root@966ec86e7af3:~/bin# chmod 700 *
chmod: missing operand after '700*'
Try 'chmod --help' for more information.
root@966ec86e7af3:~/bin# chmod 700 *
root@966ec86e7af3:~/bin# ls -l
total 8876
-rwx----- 1 root root 1115336 Mar 12 05:05 bash
-rwx----- 1 root root  38984 Mar 12 05:05 bunzip2
-rwx----- 1 root root  38984 Mar 12 05:05 bzcata
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzcmp -> bzdiff
-rwx----- 1 root root  2227 Mar 12 05:05 bzdiff
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzegrep -> bzgrep
-rwx----- 1 root root  4877 Mar 12 05:05 bzexe
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzfgrep -> bzgrep
-rwx----- 1 root root  3641 Mar 12 05:05 bzgrep
-rwx----- 1 root root  38984 Mar 12 05:05 bzip2
-rwx----- 1 root root 14328 Mar 12 05:05 bzip2recover
lrwxrwxrwx 1 root root    6 Mar 12 05:05 bzless -> bzmore
-rwx----- 1 root root  1297 Mar 12 05:05 bzmore
-rwx----- 1 root root 43744 Mar 12 05:05 cat
```

图 3. 修改文件权限

3. 实验 2 步骤 3 中令修改 bash 文件的 SUID 和 SGID 权限

使用命令 `chmod u-w bash`，修改 bash 文件的 SUID 权限，使得用户不能再对 bash 文件做写入操作。可以观察到权限由 `-rwx-----` 变为了 `-r-x-----`。

使用命令 `chmod g+x bash`，修改 bash 文件的 SGID 权限，添加群组用户可以执行 bash 文件的权限。可以观察到权限由 `-r-x-----` 变为了 `-r-x--x---`。

使用命令 `chmod o+r bash`，添加权限使得其他用户可以读 bash 文件。可以观察到权限由 `-r-x--x---` 变为了 `-r-x--xr--`。

```

-rwx----- 1 root root 1983 Mar 12 05:05 zcat
-rwx----- 1 root root 1677 Mar 12 05:05 zcmp
-rwx----- 1 root root 5879 Mar 12 05:05 zdiff
-rwx----- 1 root root 29 Mar 12 05:05 zegrep
-rwx----- 1 root root 29 Mar 12 05:05 zfgrep
-rwx----- 1 root root 2080 Mar 12 05:05 zforce
-rwx----- 1 root root 7589 Mar 12 05:05 zgrep
-rwx----- 1 root root 2205 Mar 12 05:05 zless
-rwx----- 1 root root 1841 Mar 12 05:05 zmore
-rwx----- 1 root root 4552 Mar 12 05:05 znew
root@966ec86e7af3:~/bin# chmod u-w bash
root@966ec86e7af3:~/bin# ls -l bash
-r-x----- 1 root root 1115336 Mar 12 05:05 bash
root@966ec86e7af3:~/bin# chmod g+x bash
root@966ec86e7af3:~/bin# ls -l bash
-r-x--x--- 1 root root 1115336 Mar 12 05:05 bash
root@966ec86e7af3:~/bin# chmod o+r bash
root@966ec86e7af3:~/bin# ls -l bash
-r-x--xr-- 1 root root 1115336 Mar 12 05:05 bash
root@966ec86e7af3:~/bin#

```

图 4.修改 bash 文件的 SUID 和 SGID 权限

4. 实验 3 步骤 1 中 GUEST 账户登陆失败；

由图 5 可以看到，在实验前可以使用 GUEST 账户登录成功。

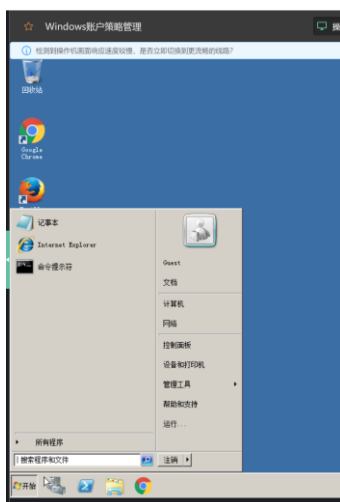


图 5.GUEST 账户正常登录界面

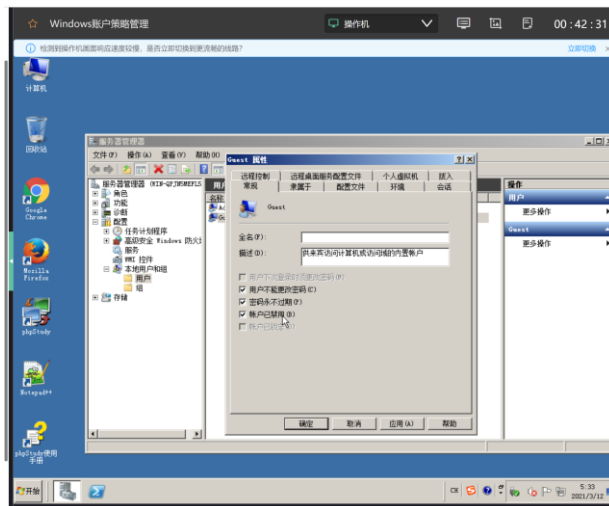


图 6. 账户禁用选项

然后单击 GUEST 用户，选择属性，并打勾上“账户已禁用”，如图 6 所示。

再次选择切换用户，此时发现 GUEST 用户无法登录，屏幕显示“您的账户已被停用，请向系统管理员咨询”，如图 7 所示。

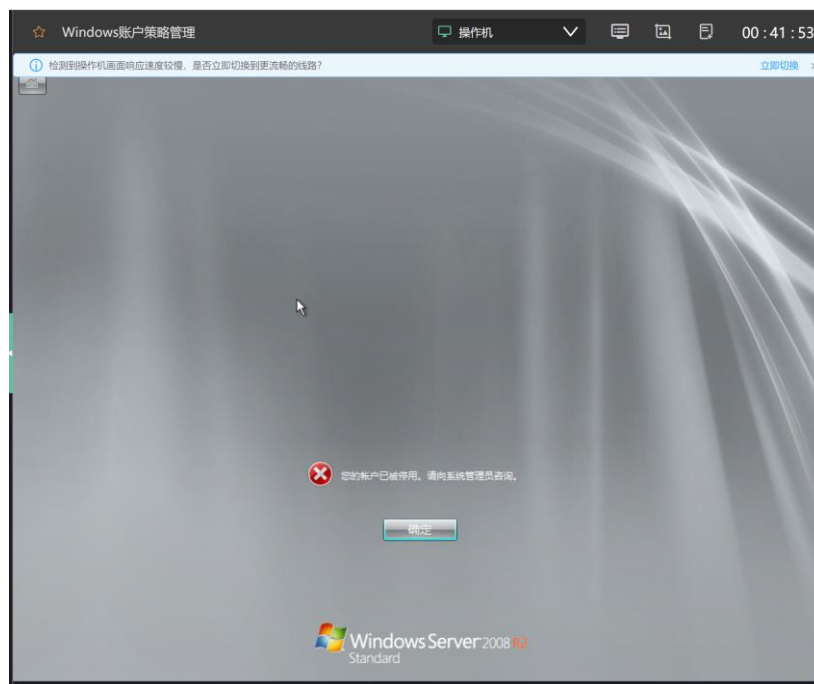


图 7. 账户被停用界面

5. 实验 3 步骤 3 中的证书信息与最后保存的证书文件；
- 在控制台里选择添加管理单元，并添加证书，选择导出证书，并设置用以保护证书私钥的密码。可以看到导出的证书信息，如图 8 所示。

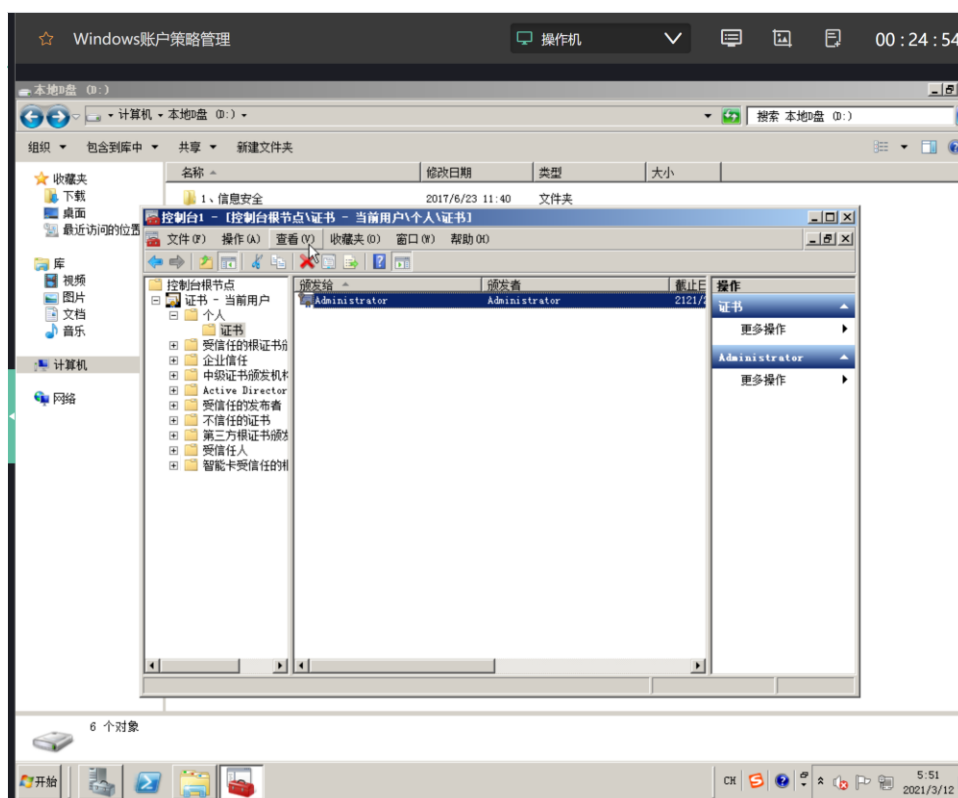


图 8. 导出证书信息

最后保存的证书文件如图 9 所示。

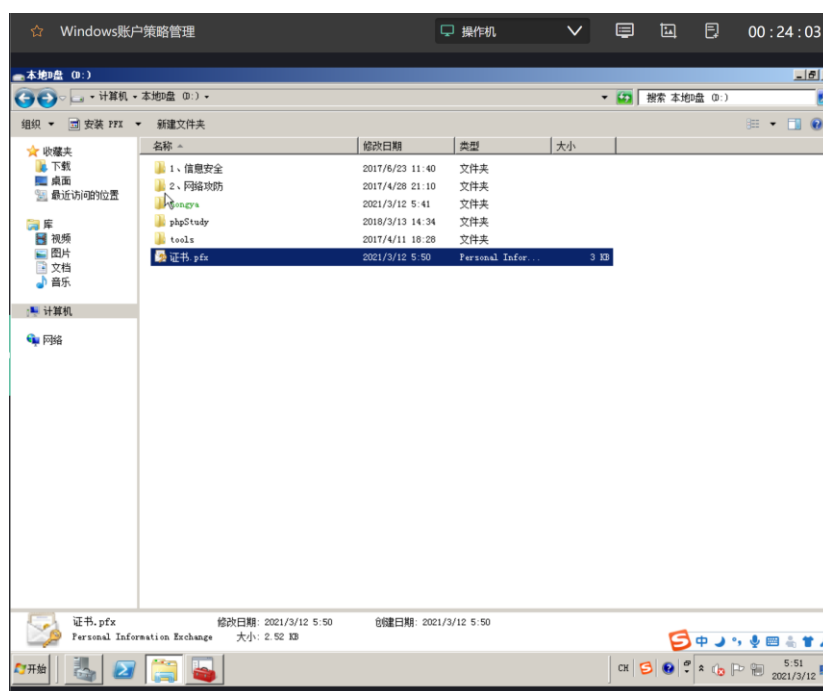


图 9. 证书文件

6. 实验 4 步骤 2 中审核成功与失败的事件属性。

在用户策略里设置好审核策略，然后注销当前用户，重新登录，第一次登录时输错密码，第二次再输入正确的密码，进入系统后点开“安全”，在审核记录中可以看到显示为“登录”任务的审核事件，可以看到其中有一条审核失败的记录，之后是审核成功的事件。具体事件属性如下图所示：



图 10. 审核失败事件属性 1



图 11. 审核失败事件属性 2

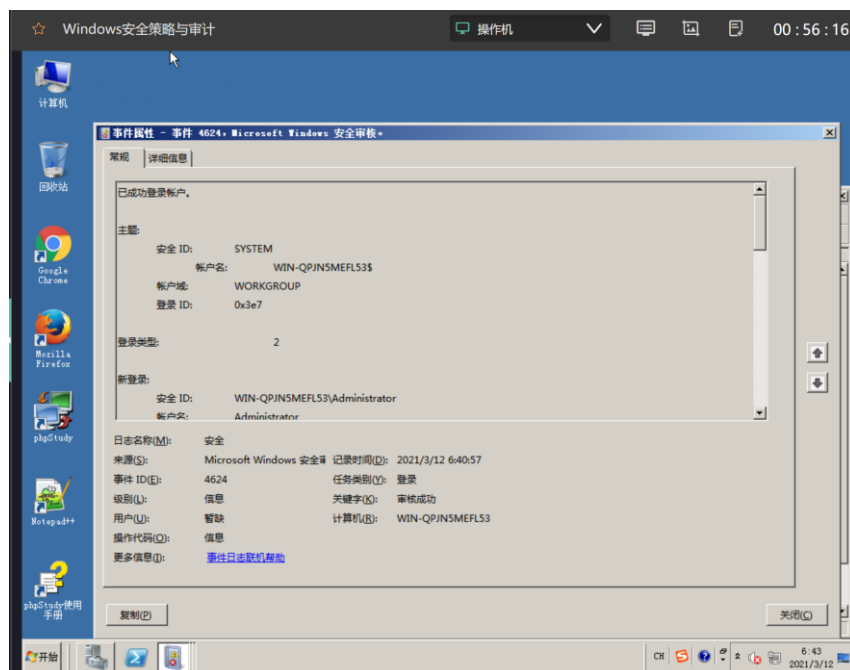


图 12. 审核成功事件属性 1

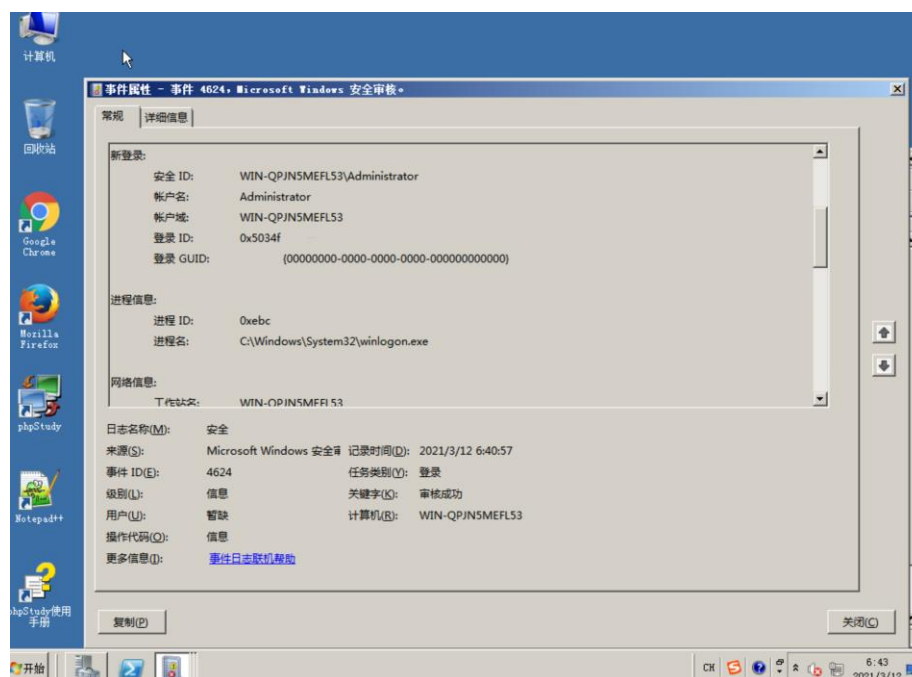


图 13. 审核成功事件属性 2

四、分析和思考（60 分）

1. 若 bash 文件的权限为“r-x--xr--”，其含义是什么，用数字表示该文件的权限应为多少？对于一个普通文本文件和一个机密文件，为保证实用性与安全性，分别设置怎样的权限较为合理，为什么？（15 分）

答：修改文件权限的命令共有 10 个参数，第 1 个参数属于管理员，而后 9 个参数与 chmod 直接关联。其中，第 2-4 个参数属于 user，第 5-7 个参数属于 group，第 8-10 个参数属于 other，而 r、w、x 分别表示可读、可写和可执行。

所以“r-x--xr--”权限表示文件所有者可读、可执行，同组用户可执行，其他用户可读的模式。用数字表示应为 514。

普通文件的密级要求不高，为便捷用户使用，保证实用性，可以按照应用场合定义权限。一般可以设置为：文件所有者拥有读、写、执行的权限，群组用户拥有读、执行的权限，而其他用户只可拥有读的权限，即“rwxr-xr--”，数字表述为 754。

机密文件有严格的保密要求，为保证安全性，一般需要按照强制存取控制和自主存取方法来确定文件密级，高许可证密级主体不可写入更新低密级对象，只有大于客体密级的主体才能读取客体的信息。为了实现机密性，应设置为文件所有者拥有读、写、执行的权限，而群组用户和其他用户没有权限，即“rwx-----”，数字表述为 700。

2. 在 windows 环境下，利用 EFS 服务实现或验证以下功能：
- 将未加密的文件转移到加密的文件夹中，即可提升其安全性
 - EFS 可加密任意类型文件，也可加密任意文件夹，无论其处于何位置
 - 经 EFS 加密的文件转移到 U 盘中后，不会降低其安全性
 - 帐户 A 进行 EFS 加密的文件/文件夹无法用帐户 B 打开，除非设置共享
 - 只有对加密文件进行共享设置后，其它用户方可对该文件进行读取、执行或删除操作
 - 帐户 A 注销后，只要重新注册同名且同性质的账户就可以打开原帐户 A 用 EFS 加密的文件/文件夹

说明上述实验的结果，并总结 EFS 的特点。（30 分）

答：a) 答：功能正确。转移后文件图标变化，未加密的文件转移到加密的文件夹后也变为了加密文件。

b) 答：功能错误。EFS 只能加密有权限读写的文件，并且加密只针对 NTFS 盘。EFS 可以加密 txt、jpg、doc、pdf 等文件格式的文件，加密文件夹时可选择加密子文件夹及目录下文件，也可选择不加密父文件夹。

c) 答：功能正确。经过 EFS 加密后的文件转移到 U 盘后仍显示加密图标，仍是加密状态，不会降低安全性。

d) 答：**功能正确**（但在共享时必须添加账户 B 的证书）。账户 B 点击访问加密的文件时，弹出“拒绝访问”的窗口。若账户 A 设置了共享，并在文件的安全属性中添加了账户 B 的证书，再登录账户 B 即可进行权限许可的操作。

e) 答：**功能错误**。若账户 A 只是对加密文件夹设置了共享，其他用户仍然无法访问。账户 A 还需要在文件的安全属性中添加账户 B 的授权证书，再登录账户 B 即可进行读取、执行、删除等操作。

f) 答：**功能错误**。即使注销并重新注册同名、同性账户也无法打开原账户加密的文件，EFS 加密的安全性得到验证。

总结：EFS 服务的特点：

- 1、对授权用户完全透明，用户访问被加密的文件夹时不需要额外操作，而非授权用户访问时才会显示拒绝访问；
- 2、便于取消加密，只需要在文件夹的高级属性窗口，取消“加密内容以便保护数据”的勾选，确定即可。
- 3、与操作系统（在本实验中即 Windows）紧密结合，不需要再安装额外的加解密软件，使操作更加方便；
- 4、无法加密 FAT 和 FAT32，只对 NTFS 文件管理系统/分区起作用；
- 5、如果没有备份证书，重装系统后或者证书丢失后，用 EFS 加密过的文件将无法打开。

3. 安全策略审计实验中审核了事件的哪些信息？设置安全策略与审计的过程中，哪些步骤体现了信息安全的 CIA 三要素？（15 分）

答：审核了事件的**任务类别、关键字、记录时间、事件 ID、级别、账户名、账户域、登录 ID、失败原因**（包括失败原因和状态）、**进程信息、网络信息**（包括：工作站名、源网络、源端口）等信息。

信息安全的 CIA 三要素包括**保密性、完整性、可用性**。在设置安全策略与审计实验的过程中，我们共完成了**设置账户策略的密码最小长度、账户锁定阈值、审核对象访问的成功和失败、审核账户管理的成功与失败**等过程。其中，**设置最小密码长度和锁定阈值**体现了保密性，**启动审核对象访问**体现了完整性和可用性，**启动审核账户管理的成功与失败**体现了保密性和可用性。

五、实验总结（收获和心得）（5 分）

在本次实验中，我熟悉了实验平台，也通过实验一接触了 Linux 的基本操作，对于 Linux 系统上的一些命令有了初步认识。

在实验中我接触到了 Linux 的文件管理方式，学会了设置文件的所有者、修改文件权限、修改 SUID 和 SGID 权限等操作方式。同时我对 Windows 下的

账户管理策略和证书也有了一定了解，尝试了共享和证书授权等工作模式。

在完成课后思考题的过程中，我使用 Windows 10 的虚拟机实际操作了 EFS 加密实验。在 EFS 的实验中，我对 Windows 系统的多用户之间的一些操作和存储管理更加了解，刚开始我还不熟悉多用户之间文件传输，以为只要放在特定文件夹，设置共享就可以实现多用户访问 EFS 加密文件。根据实验要求，我一步步尝试并查询相关资料之后，才知道要提供被授权用户的证书并且在授权用户中对其信任，才能实现其他用户对加密文件的访问。

在实验过程中我体会到了文件管理、账户管理和证书授权的重要性，这些都是操作系统安全的基础，同时也启示我授权、证书的重要性。在现实中证书也在方方面面保障着我们的安全，这吸引我更认真地去了解、学习现实安全架构中证书的颁发、授权、分发的细节。

六、尚存问题或疑问、建议（5 分）

- 1、在实验中，虚拟机提示“当前卡顿，可以切换至新的线路”，点击后虽然没有卡顿现象了，但是在实验中启用 GUEST 却未显示切换用户。这是由于切换线路实际上是更换了镜像的过程，与实验要求的环境有所出入，无法正常完成实验。

解决方案：重启计算机即可，注意不要再次切换线路。

- 2、在课后完成思考题时，我使用自己的电脑，在虚拟机上实现 EFS 加密实验，起初我无法对 U 盘文件进行操作，非常困惑。后来查阅了 EFS 加密的相关实验要求后发现，EFS 加密实验要求文件系统和分区是 NTFS 的，而我使用的 U 盘是 FAT32 的，所以无法完成指定实验操作。

解决方案：我使用了另一个 NTFS 的 U 盘就能正常完成实验要求了。