



SINGAPORE POLYTECHNIC

SCHOOL OF DIGITAL MEDIA & INFOCOMM TECHNOLOGY  
DIPLOMA IN INFOCOMM SECURITY MANAGEMENT

## SLIN Assignment

*David Zhu*  
*1703177*

supervised by  
Mr Karl Kwan

July 26, 2018

# Contents

<b>1</b>	<b>Gaining Access</b>	<b>3</b>
1.1	Recovering root password . . . . .	3
1.2	Recover user passwords . . . . .	4
1.3	Grub Password . . . . .	4
<b>2</b>	<b>Setting IP address</b>	<b>4</b>
<b>3</b>	<b>Changing login prompt message</b>	<b>5</b>
<b>4</b>	<b>Editing Web Root</b>	<b>6</b>
<b>5</b>	<b>User Permissions</b>	<b>7</b>
<b>6</b>	<b>Samba Setup</b>	<b>7</b>
<b>7</b>	<b>Accessible Home Directory</b>	<b>8</b>
<b>8</b>	<b>Department C Web Page Forbidden</b>	<b>9</b>
<b>9</b>	<b>Crontab</b>	<b>10</b>
<b>10</b>	<b>FTP setup</b>	<b>10</b>
<b>11</b>	<b>FTP logging</b>	<b>10</b>
<b>12</b>	<b>PATH &amp; SSH &amp; XINETD</b>	<b>11</b>
12.1	Fixing Date Command . . . . .	11
12.2	Fix SSH . . . . .	11
12.3	Disable tftp . . . . .	11
<b>13</b>	<b>Firewall setup</b>	<b>12</b>
<b>14</b>	<b>Security Recommendations</b>	<b>12</b>
14.1	Nginx Server Token . . . . .	12
14.2	SSH Key login . . . . .	12
14.3	NTP . . . . .	13
<b>15</b>	<b>Misc</b>	<b>13</b>

## List of Figures

1	Reset root password . . . . .	3
2	Login Success . . . . .	4
3	Grub Credentials . . . . .	4
4	Output of nmcli con show . . . . .	5
5	RHEL Version . . . . .	5
6	New Login Prompt . . . . .	5
7	CBA Website . . . . .	6
8	Possible Web-roots . . . . .	6
9	ABC Company Web Roots . . . . .	6
10	/var/pages . . . . .	6
11	/usr/share/nginx/html . . . . .	6
12	Wrong Company . . . . .	9
13	Corrected Web page . . . . .	9
14	Content of Cron Files . . . . .	10
15	tftp . . . . .	11
16	Add firewall rules . . . . .	12
17	List firewall services . . . . .	12

## List of Tables

1	User Passwords . . . . .	4
2	xferlog synatx . . . . .	11

# 1 Gaining Access

## 1.1 Recovering root password

Usually password resets can be done by booting into single user mode, however since we do not have the grub password we can do it using an bootable ISO.

First we boot from an ISO then mount the red hat linux root partition and chroot into it, then as root user run the passwd command to change password, lastly we have to add `/.autorelabel` file to prevent SELinux errors.

```
[manjaro-i3 manjaro]# fdisk -l
Disk /dev/loop0: 63.2 MiB, 66281472 bytes, 129456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop1: 217.7 MiB, 228216832 bytes, 445736 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop2: 682.8 MiB, 715939840 bytes, 1398320 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop3: 420.1 MiB, 440537088 bytes, 860424 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x000cb401

Device      Boot      Start        End    Sectors   Size Id Type
/dev/sda1   *          2048    1026047    1024000   500M 83 Linux
/dev/sda2             1026048  41943039  40916992  19.5G 8e Linux LVM

Disk /dev/mapper/rhel-swap: 2 GiB, 2147483648 bytes, 4194304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rhel-root: 17.5 GiB, 18798870528 bytes, 36716544 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[manjaro-i3 manjaro]# ls /
bin  desktopfs-pkgs.txt  etc  lib  livefs-pkgs.txt  opt  root  lib  run  srv  tmp  var
boot dev             home lib64 mnt      proc  rootfs-pkgs.txt  sbin  sys  usr

[manjaro-i3 manjaro]# mount /dev/mapper/rhel-root /mnt
[manjaro-i3 manjaro]# chroot /mnt/
[root@manjaro-i3 /]# passwd
Changing password for user root.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@manjaro-i3 /]# touch /.autorelabel
[root@manjaro-i3 /]#
```

Figure 1: Reset root password

Finally we can reboot to RHEL and login with our new password.

```
Welcome to server!
IP 192.168.94.23/24

harry hee ha

server login: root
Password:
Last login: Sun Jul 15 23:54:42 on tty1
[root@server ~]# _
```

Figure 2: Login Success

## 1.2 Recover user passwords

After logging in as root we can retrieve `/etc/shadow` and crack the password hashes with hashcat or any other password cracking tool.

User	Password
harry	hahaha
bob	bob123
john	john123
bill	bill123

Table 1: User Passwords

## 1.3 Grub Password

If we look at what's in harry's home directory, we see an interesting file named `grub.txt` which contains "*Grub user in 01\_users file.*" If we find the `01_users` file we will get the credentials for grub which is **user: hansel, password: gretel**.

```
[root@server ~]# find / -name 01_users -type f -exec cat {} \;
cat << EOF
set superusers="hansel"
password hansel gretel
EOF
```

Figure 3: Grub Credentials

# 2 Setting IP address

First we have to get the current connections and devices.

```
[root@server ~]# nmcli con show
NAME                UUID                                TYPE      DEVICE
eno16777736         13756690-ac77-b776-4fc1-f5535cee6f16  802-3-ethernet  eno16777736
[root@server ~]#
```

Figure 4: Output of nmcli con show

Then we have to figure out the version of RHEL the server is running, because the syntax for nmcli is different depending on the version.

```
[root@server ~]# cat /etc/os-release | grep VERSION
VERSION="7.0 (Maipo)"
VERSION_ID="7.0"
REDHAT_BUGZILLA_PRODUCT_VERSION=7.0
REDHAT_SUPPORT_PRODUCT_VERSION=7.0
[root@server ~]#
```

Figure 5: RHEL Version

Now that we know the RHEL version we can modify the current connection to a new static IP using nmcli. We will give the server IPv4 address of 172.16.180.100/24.

```
> nmcli con mod eno16777736 ipv4.addresses "172.16.180.100/24 172.16.180.1" ipv4.method manual
> nmcli con up eno16777736
```

### 3 Changing login prompt message

Edit `/etc/issue` to the following

Legal Notice

Unauthorized access is strictly prohibited and will be investigated. Your activities may be monitored and logged.

```
> printf 'Legal Notice\n\nUnauthorized access is strictly prohibited and will be investigated.\n\nYour activities may be monitored and logged.\n\n\n' > /etc/issue
```

Legal Notice

Unauthorized access is strictly prohibited and will be investigated. Your activities may be monitored and logged.

server login:

Figure 6: New Login Prompt

## 4 Editing Web Root

If we browse to the web server, we realize the web page does not belong to ABC Company, so we have to find ABC Company's web root and change the nginx configuration.

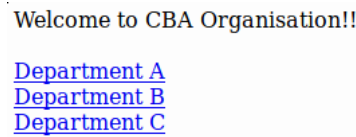


Figure 7: CBA Website

First we have to find all the web roots.

```
[root@server ~]# find / -name DeptA -type d | sed 's/DeptA//' > webroots.txt
[root@server ~]# cat webroots.txt
/var/www/html/
/var/www/
/var/pages/
/usr/share/nginx/html/
/usr/local/nginx/html/
/usr/pages/html/
```

Figure 8: Possible Web-roots

Now we search for the web roots that contain "ABC" in their index.html.

```
[root@server ~]# while read p; do grep -H ABC "$p/index.html"; done < webroots.txt
/var/pages//index.html:Welcome to ABC Organisation!!<br><br>
/usr/share/nginx/html//index.html:Welcome to ABC Organisation!!<br><br>
[root@server ~]#
```

Figure 9: ABC Company Web Roots

Now we check what's in the 2 directories.

```
[root@server ~]# ls -la /var/pages/
total 8
drwxr-xr-x. 5 root root 73 May 20 2016 .
drwxr-xr-x. 25 root root 4096 Jul 18 04:32 ..
drwxr-xr-x. 2 root root 6 May 20 2016 cgi-bin
drwxr-xr-x. 2 root root 57 May 20 2016 DeptA
drwxr-xr-x. 2 root root 23 May 20 2016 DeptB
lrwxrwxrwx. 1 root root 15 May 20 2016 DeptC -> /webpages/DeptC
-rw-r-----. 1 root root 249 May 20 2016 index.html
```

Figure 10: /var/pages

```
[root@server ~]# ls -la /usr/share/nginx/html/
total 12
drwxr-xr-x. 4 root root 95 Nov 10 2016 .
drwxr-xr-x. 3 root root 17 Nov 10 2016 ..
-rw-r--r--. 1 root root 537 Oct 19 2016 50x.html
drwxr-xr-x. 2 root root 57 Nov 10 2016 DeptA
drwxr-xr-x. 2 root root 23 Nov 10 2016 DeptB
lrwxrwxrwx. 1 root root 15 Nov 10 2016 DeptC -> /webpages/DeptC
-rw-r--r--. 1 root root 178 Nov 10 2016 index.html
-rw-r--r--. 1 root root 612 Oct 19 2016 index.html.bak
```

Figure 11: /usr/share/nginx/html

Since `/usr/share/nginx/html` has been modified more recently, we will be using it as the web root and add footer to all the default web pages.

```
> perl -pi -e 's/\usr/pages/html/\usr/share/nginx/html/' /etc/nginx/conf.d/default.conf && \
> systemctl reload nginx
> find -L /usr/share/nginx/html -type f -name index.html \
> -exec sh -c 'echo \<br>ABC Company, Singapore >> "$1" -- {} \;
```

```
> semanage fcontext -a -t httpd_sys_content_t "/usr/share/nginx/html(/.*)?"
> restorecon -Rv /usr/share/nginx/html
```

## 5 User Permissions

Create a new group DeptA and add bob and bill

```
> groupadd DeptA
> usermod -a -G DeptA bob
> usermod -a -G DeptA bill
```

Change the group of **/usr/share/nginx/html/DeptA** to DeptA and set permissions so that all files and folders created in DeptA will belong to the DeptA group and users in the DeptA group can edit all the files in DeptA.

```
> chgrp -R DeptA /usr/share/nginx/html/DeptA
> chmod 2775 /usr/share/nginx/html/DeptA
> chmod -R 664 /usr/share/nginx/html/DeptA/*
> setfacl -d -m group:DeptA:rw /usr/share/nginx/html/DeptA
```

Password of bob and bill can be found in Table 1.

## 6 Samba Setup

In order to allow access to both linux and windows computers, we have to setup a samba service on the server. Since the samba package is already installed, we just need to add the following to the bottom of **/etc/samba/smb.conf**

```
[DeptA-Web]
comment = Web Files for Department A
path = /usr/share/nginx/html/DeptA
valid users = bob bill
public = no
browsable = no
writable = yes
printable = no

# changes the permissions to rw-rw-r-
create mask = 0664
force create mode = 0664

# Change directory permissions to rwxrwsr-x
directory mask = 2775
force directory mode = 2775
```



Remove or comment out the following from `/etc/samba/smb.conf` and run `testparm` to check the configuration.

```
[homes]
    comment = Home Directories
    browseable = no
    writable = yes
;    valid users = %S
;    valid users = MYDOMAIN\%S
[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = no
    guest ok = no
    writable = no
    printable = yes
```

The following commands will create users bob and bill respectively for the samba service. The passwords of each user can be set as their respective password in Table 1.

```
> smbpasswd -a bob
> smbpasswd -a bill
```

Add DNS for server hostname to prevent errors.

```
> echo '127.0.0.1 server.example.com' >> /etc/hosts
```

Start and enable the service

```
> systemctl start smb && systemctl enable smb
```

SELinux

```
> semanage fcontext -d "/usr/share/nginx/html/DeptA(/.*)?"
> semanage fcontext -a -t public_content_rw_t "/usr/share/nginx/html/DeptA(/.*)?"
> restorecon -Rv /usr/share/nginx/html/DeptA
> setsebool -P allow_smbd_anon_write 1
```

Since the share is not browsable, the name of the share has to be given to the users before they can connect to it.

## 7 Accessible Home Directory

Remove or comment out the following block of code from `/etc/nginx/conf.d/default` which is allowing access to `public_html` folder in user home directories.

```
location ~ ^/^(.+?)(/.*)?$ {
    alias /home/$1/public_html$2;
    index index.html
    autoindex on;
}
```

Now reload nginx.

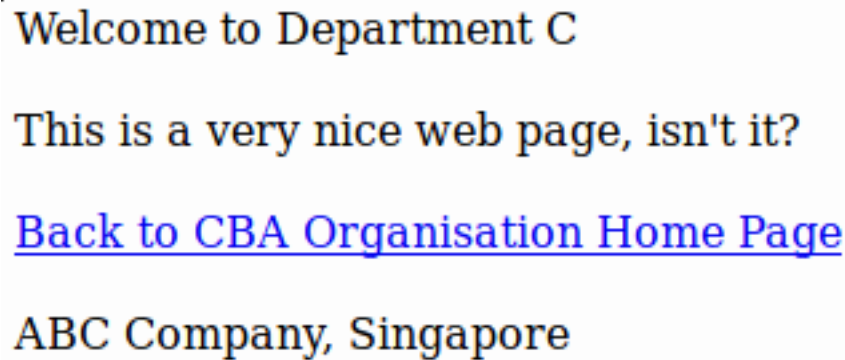
```
> systemctl reload nginx
```

## 8 Department C Web Page Forbidden

If we try to access Department C's web page, we get a 403 Forbidden. If we view the SELinux logs at `/var/log/audit/audit.log` we can see that it is being denied by SELinux. So we have to set the proper SELinux context on the files in order to display Department C's web page.

```
> semanage fcontext -a -t httpd_sys_content_t "/webpages(/.*)?"  
> restorecon -Rv /webpages
```

Now we will be able to see Department C's web page.



Welcome to Department C

This is a very nice web page, isn't it?

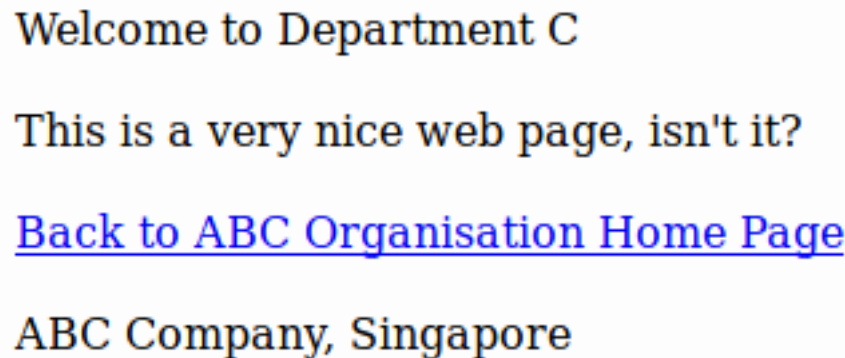
[Back to CBA Organisation Home Page](#)

ABC Company, Singapore

Figure 12: Wrong Company

However, the company name is wrongly written, since there are no other Department C on this server, we will have to correct the company name.

```
> perl -pi -e 's/CBA/ABC/' /usr/share/nginx/html/DeptC/index.html
```



Welcome to Department C

This is a very nice web page, isn't it?

[Back to ABC Organisation Home Page](#)

ABC Company, Singapore

Figure 13: Corrected Web page

## 9 Crontab

Since the logs are disappearing on an interval, it is likely a cronjob. So we need to check `/var/spool/cron/` to see the user cronjobs.

```
[root@server ~]# find /var/spool/cron/ -type f -print -exec cat {} \;
/var/spool/cron/john

* * * * * tail -2 /var/log/secure > /tmp/secure; cat /tmp/secure > /var/log/secure
/var/spool/cron/root

[root@server ~]# _
```

Figure 14: Content of Cron Files

From Figure 14 we can see that john's cronjob is trying to write the last 2 lines of `/var/log/secure` to `/tmp/secure` and write that back to `/var/log/secure`, which is effectively removing all records of `/var/log/secure` and only keeping the last 2. However due to insufficient permission on `/tmp/secure` for john, the same 2 records in `/tmp/secure` keeps getting written to `/var/log/secure` every minute. So to fix this we have to remove john's cronjob.

```
> rm -f /var/spool/cron/john
```

## 10 FTP setup

Currently users can download files from locations other than `/var/ftp/pub` because the root for anonymous users is set at the system root. We have to change root for anonymous users to `/var/ftp/pub` to restrict anonymous users access to only that directory. Set the following in `/etc/vsftpd/vsftpd.conf` and restart vsftpd

```
anon_root=/var/ftp/pub
```

```
> perl -pi -e 's/anon_root.+/anon_root=/var/ftp/pub/' /etc/vsftpd/vsftpd.conf && \
> systemctl restart vsftpd
```

## 11 FTP logging

Set the following in `/etc/vsftpd/vsftpd.conf` and restart vsftpd

```
xferlog_enable=YES
```

```
> perl -pi -e 's/xferlog_enable.+/xferlog_enable=YES/' /etc/vsftpd/vsftpd.conf && \
> systemctl restart vsftpd
```

Sample xferlog entry.

```
fri Jun 29 22:53:07 2018 1 ::ffff:172.16.180.1 45 /pub/document1 b _ o a mozilla@example.com ftp 0 * c
```

Data	Field Name	Description
fri Jun 29 22:53:07 2018	Time of transfer	The time that the transfer occurred
1	Transfer Time taken	The total time in seconds for the transfer
::ffff:172.16.180.1	Remote Host	The remote host name
45	File Size (Bytes)	The number of bytes transferred
/pub/document1	File Name	The name of the transferred file
b	Transfer Type	<b>a</b> - ascii, <b>b</b> - binary
\_	Special Action Flag	<b>C</b> - File was compressed, <b>U</b> - file was uncompressed, <b>T</b> - File was archived, <b>_</b> - No action was taken
o	Direction	<b>O</b> - outgoing, <b>i</b> - incoming
a	Access Mode	<b>a</b> - anonymous, <b>g</b> - For a passworded guest user, <b>r</b> - For a real, locally authenticated user
mozilla@example.com	Username	The local username, or if anonymous, the ID string given
ftp	Service Name	The name of the service invoked, usually ftp
0	Auth Method	<b>0</b> - None, <b>1</b> - RFC 931 authentication
*	Auth User-ID	The user ID returned by the authentication method. A * is used if an authenticated user ID is not available.
c	Completion Status	<b>c</b> - indicates complete transfer, <b>i</b> - indicates incomplete transfer

Table 2: xferlog syntax

Location of the xferlog file is at `/var/log/xferlog`

## 12 PATH & SSH & XINETD

### 12.1 Fixing Date Command

If we run *which date* as harry, we can see that it is at `/data/date` which is not the correct path of the date command. So we need to remove it from harry's PATH by commenting out or removing the `PATH=/data:$PATH` from `/home/harry/.bash_profile`.

```
> perl -pi -e 's/PATH=\data:~$PATH//' /home/harry/.bash_profile
```

### 12.2 Fix SSH

TCP Wrappers are blocking traffic to sshd, so we have to remove `sshd: ALL` from `/etc/hosts.deny`

```
> perl -pi -e 's/sshd:~sALL//' /etc/hosts.deny
```

Only allow harry to login via ssh.

```
> echo 'AllowUsers harry' >> /etc/ssh/sshd_config && systemctl reload sshd
```

### 12.3 Disable tftp

We first check if tftp is running.

```
[root@server ~]# ss -tulp | grep tftp
tcp    UNCONN    0          0          *:tftp    *:~        users:(("xinetd",1275,5))
[root@server ~]#
```

Figure 15: tftp

Set disable to yes in `/etc/xinetd.d/tftp` and reload xinetd

```
> perl -pi -e 's/((disable\s+=\s)no/$1yes/' /etc/xinetd.d/tftp && systemctl reload xinetd
```

## 13 Firewall setup

This section is done assuming the computer in Department D which will have all its traffic blocked by the server has ip of 172.16.180.50

```
> systemctl start firewalld && systemctl enable firewalld # start and enable the firewall
```

Get the default zone and add the required rules to the default zone, which in this case is *public*.

```
[root@server ~]# firewall-cmd --get-default-zone
public
[root@server ~]# firewall-cmd --zone=public --permanent --add-port=9022/tcp
success
[root@server ~]# firewall-cmd --zone=public --permanent --add-service=ftp --add-service=http --add-service=samba
success
[root@server ~]# firewall-cmd --zone=public --permanent --add-rich-rule 'rule family="ipv4" source address=172.16.180.50 drop'
success
```

Figure 16: Add firewall rules

Now we reload the firewall and check the allowed services.

```
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# firewall-cmd --zone=public --list-services
dhcpv6-client ftp http samba ssh
[root@server ~]#
```

Figure 17: List firewall services

We can remove ssh service since this is opening port 22 which we are not using as our ssh service is listening on port 9022.

```
> firewall-cmd --zone=public --permanent --remove-service=ssh
> firewall-cmd --reload
```

## 14 Security Recommendations

### 14.1 Nginx Server Token

Nginx server token should be turned off. To prevent attackers from knowing which version of nginx the server is running and searching vulnerabilities for that version of nginx. This can be done by adding *server\_tokens off;* to the http block in `/etc/nginx/nginx.conf` and reloading the nginx service.

### 14.2 SSH Key login

SSH logins should only be allowed with keys, as password logins can be brute forced quickly if a weak password is used. In general it will take a much longer time to bruteforce ssh keys as compared to passwords. This can be done by adding public keys to users' authorized key files and changing *ChallengeResponseAuthentication* and *PasswordAuthentication* to no in `/etc/ssh/sshd_config`

## 14.3 NTP

The server should use an NTP server to configure the time automatically. Things like tls certificates can have errors if the time set is not correct.

[Click here for guide](#)

## 15 Misc

Useful commands

```
> smbclient -L <ip> -U <user> # list shares
> smbclient //<ip>/<share> -U <user> # connect to share
> nginx -t # check nginx config
> testparm # check smb.conf
> sestatus -v # selinux status with file and process contexts
> sestatus -b # selinux boolean variables
> semanage fcontext -d "/path(/.*)?" # delete file context recursively
> sestatus -b | grep on$ # show all selinux boolean variables that are on
```