리눅스 보안

필수적으로 알아두어야 할 보안 내용에 대해 정리한다. 참고한 글이 레드헷 계열에 대한 내용이라 데비안 계열에서서는 적용되지 않는 내용일 수 있다.

ssh의 root 로그인 방지

대개의 배포판은 기본적으로 root계정으로 ssh의 접속을 막아 놓지만 그렇지 않을 수도 있으므로 확인한다. root의 접속은 반드시 su 명령으로 통하여 획득할 수 있도록 한다.

```
$ sudo vim /etc/ssh/sshd_config
PermitRootLogin no
```

ftp서버의 상위루트 노출 막기

ftp로 접속시 로그인계정 디렉토리 외에도 다른 계정의 디렉토리까지도 노출될 경우가 있다. 이럴경우 sql계정정보나 사이트의 모든 문서를 자유롭게 다운받을 수 있게된다. 이런 현상을 방지하기 위해 로그인 계정외는 노출을 금지시켜야 한다.

vsftp인 경우 아래 부분을 주석 처리 한다.

```
$ sudo vim /etc/vsftpd.conf
chroot_local_user=YES
```

만약 주석 해제하면 chroot_list_file에 지정한 파일에 등록된 유저에 대해서는 상위 폴더의 접근을 허락하게 된다.

ssh접속시 상위 디렉토리 노출시키지 않기

사용자계정들이 들어가게 되는 디렉토리의 퍼미션을 변경한다. 이럴경우 ftp접속시에도 상위디렉토리가 노출되지 않는 효과도 생긴다. 만약 /home 디렉토리에 사용자계정 디렉토리들이 들어가게 된다면..

```
$ chmod 701 /home
$ chmod 701 /
```

등으로 일반 사용자들에게 노출되기 싫은 디렉토리의 권한을 잡아주면 된다.

특정 파일을 root만 사용하게 만들기

gcc나 gcc+등과 같이 컴파일러나 df같이 하드디스크의 정보나 ps와 같이 메모리에 상주된 프로그램등, 시스템에 관련된 파일들을 일반계정 사용자가 실행할 수 없게 만들어 주면 보안상 좀 더 유리해진다.

```
$ chmod 100 /usr/bin/gcc /usr/bin/g++
$ chattr +i /usr/bin/gcc /usr/bin/g++
```

이와 같이 퍼미션을 주게 되면 일반유져들은 실행을 할수 없게 되고 +i옵션을 주어 수정,복사,삭제가 불가능하게 된다.

```
$ chmod 100 /bin/ps
$ chattr +i /bin/ps
```

이런식으로 일반계정 사용자가 실행하면 좀 껄끄러운 파일들을 루트사용자만 사용할 수 있게끔 설정해주면 된다. 특히 c나 c++컴파일러는 불순한 의도를 가지고 사용자가 사용할 수 있으므로 웹서비스만 하는 서버에서는 막아주는것이 좋다.

su명령어를 특정 사용자만 사용하게 만들기

불량 계정접속자들의 루트로그인을 막기 위해 su명령을 신뢰할 수 있는 계정 사용자에게 줄 필요성이 있다. 이 방법은 정해진 계정으로만 접속해야 만 루트사용자로 로그인을 할 수 있게 된다.

```
$ vim /etc/group
wheel:x:10:root,manager
```

그룹설정 파일을 보면 wheel:x:10:root항목이 있다. 뒤에 , su명령어로 루트접속이 가능하게 할 계정을 입력해준다. 위의 경우 manager계정만 su명령어로 루트에 접속할 수 있게된다.

그룹파일을 수정한 후 su 명령에 대해 다음과 같이 설정한다.

```
$ chown root.wheel /bin/su
$ chmod 4750 /bin/su
$ chattr +i /bin/su
```

su명령어를 root와 wheel두 그룹만 소유하게 된다.

ping차단하기

고전수법이지만 ping명령어로 서버의 속도를 저하시킬 수 있다. 그러므로 ping명령을 차단시킬 필요가 있다.

```
$ vi /etc/sysctl.conf
net.ipv4.icmp_echo_ignore_all=1
```

net.ipv4.icmp_echo_ignore_all=1를 해주면 ping명령이 차단된다. 콘솔상에서

```
$ /sbin/sysctl -w net.ipv4.icmp_echo_ignore_all=0
```

하면 다시 ping이 허용되며 0대신 1을 넣어주면 다시 차단된다.

SYN Flooding 공격방어

이것은 Dos공격의 일종으로 통신상 일어나는 현상을 교묘히 이용하여 생기는 공격이다. 이 공격을 막는 방법으로는 백로그큐를 늘려주거나 tcp_syscookies값을 1을 주어 공격을 차단할 수 있다.

```
$ vi /etc/sysctl.conf
net.ipv4.tcp_syscookies=1
```

하지만 너무 과도하게 이 방법만 믿으면 안된다. KLDP같은 포럼에서 신종 공격에 대비하는 비책들을 항상 스크랩하고 자신의 서버에 적용시키자.

SetuID 속성파일들의 권한 변경

좀 약간 헷갈리는 내용인데 Setuid로 설정된 파일은 실행시키면 실행자의 권한이 아니라 소유자의 권한으로 파일이 구동된다.

이해를 쉽게 하자면 passwd명령은 소유자가 루트이지만 모든 계정사용자가 자신의 비밀번호를 바꿀때 사용할 수 있다. 하지만 이 passwd는 사용자계정으로 실행해도 root권한으로 실행되게 된다.

```
$ ls -al /usr/bin/passwd
-rwsr-xr-x 1 root root 22984  1월  7  2007 /usr/bin/passwd
```

위와같이 rwsr부분이 rwxr이 아니라 rwsr이 된다. 이것이 바로 setuid가 설정된 파일이라는 것이다.

시스템 관리자의 패턴을 잘 아는 불순한자들이라면 setuid(0)과 같은 c언어 몇줄로서 백도어 프로그램을 시스템에 숨겨놓고 Setuid파일로 위장해서 루트권한을 획득할 수 있다. 그러므로 setuid로 정의 된 파일들을 항상 관심있게 체크해볼 필요가 있다.

find 명령으로 setuid가 부여된 파일들중 잘 쓰지않는 파일들을 찾아 권한을 변경한다.

```
$ find / -user root -perm -4000 -print
/usr/kerberos/bin/ksu
/usr/lib/nspluginwrapper/plugin-config
/usr/libexec/openssh/ssh-keysign
/usr/sbin/userhelper
/usr/sbin/usernetctl
/usr/sbin/suexec
/usr/sbin/ccreds_validate
/usr/bin/chfn
/usr/bin/rcp
/usr/bin/newgrp
/usr/bin/rlogin
/usr/bin/sudoedit
/usr/bin/at
/usr/bin/rsh
/usr/bin/chsh
/usr/bin/chage
/usr/bin/gpasswd
```

```
/usr/bin/crontab
/usr/bin/staprun
/usr/bin/sudo
/usr/bin/passwd
/usr/bin/Xorg
/lib/dbus-1/dbus-daemon-launch-helper
/sbin/umount.nfs4
/sbin/pam_timestamp_check
/sbin/mount.ecryptfs_private
/sbin/mount.nfs
/sbin/unix_chkpwd
/sbin/mount.nfs4
/sbin/umount.nfs
/bin/su
/bin/ping6
/bin/umount
/bin/mount
/bin/ping

$ find / -user root -perm -2000 -print
/sbin/netreport
/usr/bin/wall
/usr/bin/crontab
/usr/bin/ssh-agent
/usr/bin/write
/usr/bin/lockfile
/usr/bin/screen
/usr/local/firewall
/usr/libexec/utempter/utempter
/usr/sbin/sendmail.sendmail
/usr/sbin/lockdev
```

이런식으로 setuid파일들을 검색하고 ping과 같이 꼭 root권한이 필요없는 파일들이나 일반유져들에게 없어도 되는 파일들은 setuid속성을 없애준다. 방법은 간단히

```
$ chmod 100 /bin/ping
```

요런식으로 권한을 바꾸어주면 된다.

```
/usr/bin/change
/usr/bin/wall
/usr/bin/chfn
/usr/bin/at
/bin/mount
/bin/unmount
/usr/bin/crontab
/usr/bin/newgrp
/usr/bin/write
/usr/sbin/usernetctl
/bin/ping
```

```
/bin/traceroute
```

위의 파일들은 일반계정사용자에게는 전혀 필요없고 또한 서버운영상 불필요한 것들도 있으므로 권한을 바꾸어준다. 그 외 나머지 파일들은 파일의 크기등을 따로 보관후 쉘등을 이용해서 일괄적으로 용량등을 체크해서 파일의 변화가 있었는지 주시할 필요가 있다.

그 외...

ftp와 ssh같은 서버들의 포트를 바꾸는것도 좋다. lastlog등으로 항상 ssh접속을 시도한 불순한자들을 찾아내고 root의 .bash_history파일이 용량이 적어졌다거나 하면 불순한자들이 있을지도 모른다는 생각을 갖자.

참고

- 리눅스 기초 보안처리