



Honeynet Maintenance Procedures and Tools

Carlos Henrique P C Chaves
Lucio Henrique Franco
Antonio Montes

`{carlos.chaves,lucio.franco,antonio.montes}@cenpra.gov.br`



Honeynet.BR



Outline

- Discussion
- Honeypot Classification
- Developed Procedures
- Developed Tools
- Conclusion
- Contact



Discussion

- Honeynets... What 're they for?
- Frequent operation: deactivate, collect information, clean up and put back in operation.
- Honeypot redeployment is time consuming.
- Procedures to help in maintenance tasks.
- Development of a Honeypot Deployment CD-ROM.



Honeypot Classification

- Security resources with no production value.
- Any observed activity is suspected by default.
- High-interaction honeypots.
 - Research tool.
- Low-interaction honeypots.
 - Fake services and operating systems.
 - Complement IDS in production networks.



Developed Procedures

- Several honeypots with different configurations.
- Automate the procedures to reduce deployment time.
- Database for the storage of information about the honeypots.
- Simplify the process of gathering information for status reports.



Developed Procedures

- Honeypot Deployment Procedure
 - Definition of the operating system and services.
 - 1. Write the hard disk with constant data (0).
 - 2. Install the operating system.
 - 3. Install, configure and initialize services.
 - 4. Execute the honeypot configuration script.
 - 5. Configure keystrokes logging system.
 - 6. Execute script to generate SHA1 and MD5 file hashes.
 - 7. Execute script to generate the system's status.
 - 8. Store the SHA1 and MD5 hashes and the system's status in a loghost.



Developed Procedures

- Honeypot Deployment Procedure
 - 9. Erase all traces of steps 5-8.
 - 10. Generate the system 's image and export it to the loghost.
 - 11. Connect the honeypot in the honeynet and start monitoring it.



Developed Procedures

- Post-deployment Procedure
 - Create a standardized deployment log for each honeypot.
 - HN-[org]-DEPLOY-[date].
 - Send it by e-mail to project members.
 - Update the log file every time a honeypot is compromised and deactivated.



Developed Procedures

HONEYNET-BR SUMMARY:

GenII Honeynet with a class C IP address block exposed directly to the Internet. Typical small organization /24 network.

IP RANGE: xxx.xxx.xxx.1-254

ACTIVE HONEYPOTS:

System Name: Hp1

System IP: 10.0.0.36

System OS: RH 7.2, default, no patches

Date Deployed: 2004, Nov 24

Time Deployed: 10:50am

Responsible: Lucio Henrique Franco

System Summary:

- minimal installation, with portmap(4.0-38), with xinetd(2.3.3-1)
- kernel version: 2.4.7-10

Modifications to System:

- installed ntpd(4.1.0-4), httpd(1.3.20-16 with ssl & with mod_perl 1.2 4_01-3), php(4.0.6-7), wu-ftp(2.6.1-18)
- installed static bash patch
- installed smartl
- add default accounts

Status Change:

HONEYNET-BR SUMMARY:

GenII Honeynet with a class C IP address block exposed directly to the Internet. Typical small organization /24 network.

IP RANGE: xxx.xxx.xxx.1-254

ACTIVE HONEYPOTS:

System Name: Hp1

System IP: 10.0.0.36

System OS: RH 7.2, default, no patches

Date Deployed: 2004, Nov 24

Time Deployed: 10:50am

Responsible: Lucio Henrique Franco

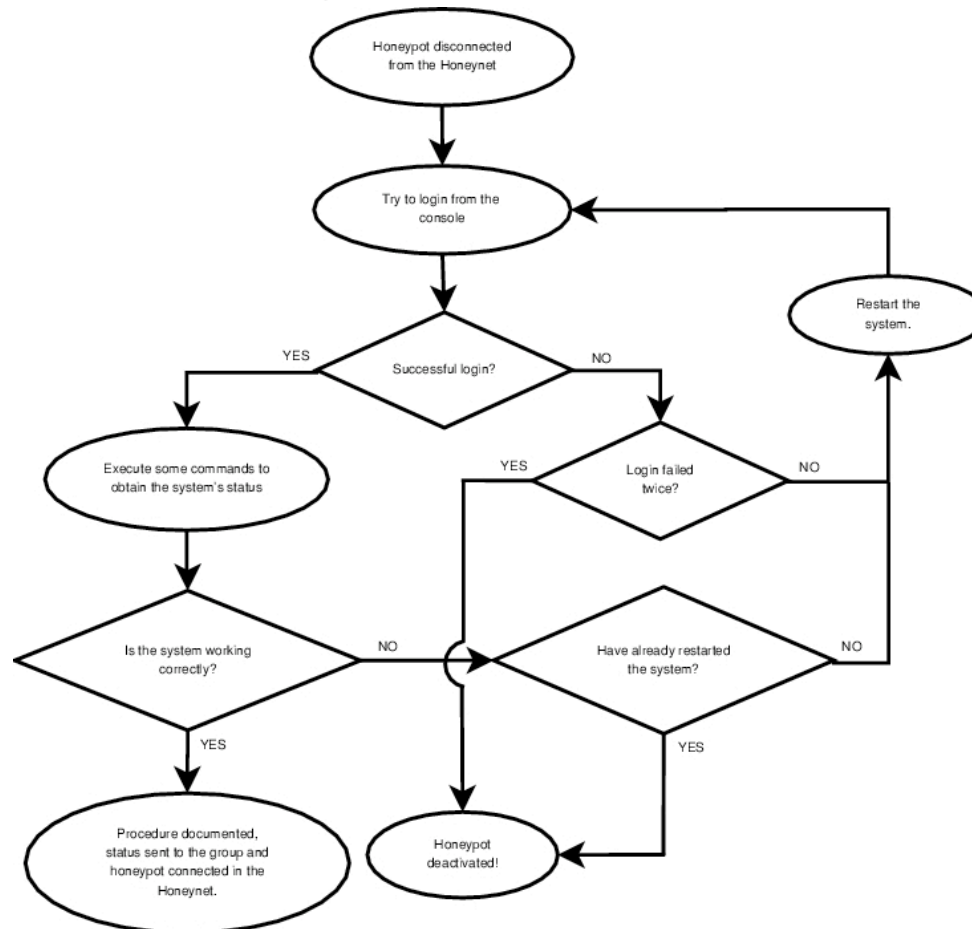
(...)

Status Change:

- First succesfull attack: Sat Dec 04 15:43:50 GMT 2004 (Resp. Cae)
 - Attack description: OpenSSL Buffer overflow exploit
 - Attack origin: 10.10.0.104
 - Whois: 10.10.0.0 - 10.10.255.255 -> Foobar Technologies -> Brazil
 - Offline on 12-10-2004 at 08:30am (Lucio)

Developed Procedures

■ Honeypot Monitoring Procedure





Developed Procedures

- Honeypot Deactivation Procedure
 - 1. Disconnect the honeypot from the Honeynet.
 - 2. Collect and analyse the system status.
 - 3. Generate the system's image and export it to the loghost.
 - 4. The honeypot is ready to be reinstalled.

- If necessary, reboot with a live CD-ROM or floppy disk of an operating system.
- Compare the initial image with the generated one.



Developed Tools

- Honeypot Deployment Script
 - Three options:
 - (a) Erase the hard disk and create a new partition.
 - (b) Erase an specified partition and use it.
 - (c) Generate and export the new system image.

 - New honeypot \Rightarrow option (a).
 - ✓ Restoring and installing the same system \Rightarrow option (b).
 - ✓ Uses SSH or Netcat to transfer the images.
 - ✓ Works with Red Hat Linux honeypots (for now!!!).



Developed Tools

- Honeypot Configuration Script
 - Two options:
 - (a) Configure a new honeypot.
 - (b) Generate the system's status, the MD5 and SHA1 hashes, clean up all traces and export the image to a loghost.
 - New honeypot \Rightarrow option (a): configure root password, network, create user's accounts, post-installation and calls honeypot cleanup script.
 - Restoring the honeypot \Rightarrow option (b): calls honeypot cleanup script.



Developed Tools

- SHA1 and MD5 Hash Generation Script
 - Many directories (input) \Rightarrow file for each directory.
 - ✓ Generated files are compressed.
 - ✓ [honeypot name]+[directory path]+[sha1|md5]
 - ✓ Runs in Linux, FreeBSD, Windows (cygwin4).



Developed Tools

■ Honeypot Cleanup Script

- Main goal: generate the MD5 and SHA1 hashes, the system's status, clean up all traces and export the image to a loghost.
- MD5 and SHA1 hashes \Rightarrow /bin, /sbin, /dev, /etc, /usr/bin, /usr/sbin, /usr/local/bin, /usr/local/sbin.
- ✓ System's status \Rightarrow ifconfig, netstat, ps, lsof, lsmod, rpcinfo, df, socklist, rpm, last \Rightarrow [cmd_name].txt
- ✓ Export the information to a loghost using SSH.
- ✓ Remove all traces using shred.



Developed Tools

- High-Interaction Honeypot Deployment CD-ROM
 - 1. Boot the honeypot from the CD-ROM ⇒ honeypot deployment script is executed.
 - 2. Honeypot's OS image ⇒ hard disk, all scripts ⇒ /root, call to honeypot configuration script ⇒ /etc/profile.
 - ▼ 3. Reboot using the honeypot's own system. Root logs in ⇒ configuration script executed.
 - ▼ 4. Reboot using the CD-ROM to generate and export the honeypot's image.
 - ▼ 5. Reboot the system again without the CD-ROM ⇒ honeypot deployed.



Developed Tools

- Operation Monitoring Script
 - Automate and standardize the honeypot's status.
 - CD-ROM with statically compiled binaries.
 - Script copied and run from a floppy disk.

 - 1. Mount the floppy containing the script.
 - 2. Execute the script using the CD-ROM mounting point and output directory as parameters.
 - 3. Output directory created.
 - 4. Commands executed \Rightarrow output appended to status file.
 - ✓ 5. CD-ROM and floppy unmounted.



Developed Tools

```
$cat /mnt/floppy/15_03_2004/hp1

### hostname ###
hp1.localdomain

### df -h ###
Filesystem      Size  Used Avail Capacity  Mounted on
/dev/sda1       1.7G  628M  1.0G   38%        /
none            46M   0     46M   0%        /dev/shm
/dev/fd0        1.4M  3.5k  1.3M   1%        /mnt/floppy
/dev/cdrom      358M  358M  0      100%       /mnt/cdrom

### w ###
12:46pm up 32 days, 0 min, 1 user, load average: 0.28, 0.42, 0.25
USER  TTY  FROM  LOGIN@  IDLE   JCPU   PCPU   WHAT
root  tty1  -     12:32pm 3.00s  4.00s  0.27s /bin/sh ./statu

### last ###
root    tty1                Tue Jul 27 12:32  still logged in
root    tty1                Fri Jun 25 12:53 - 12:54 (00:01)
reboot  system boot  2.4.7-10 Fri Jun 25 12:52      (31+23:54)
root    tty1                Wed Jun 16 10:24 - 10:25 (00:00)
reboot  system boot  2.4.7-10 Wed Jun 16 10:22      (9+02:26)

### netstat -na ###
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address   Foreign Address State
tcp          0      0 0.0.0.0:1024    0.0.0.0:*      LISTEN
(...)
```



Developed Tools

```
tcp      0      0 0.0.0.0:111      0.0.0.0:*      LISTEN
tcp      0      0 0.0.0.0:80       0.0.0.0:*      LISTEN
tcp      0      0 0.0.0.0:21       0.0.0.0:*      LISTEN
tcp      0      0 0.0.0.0:11223    0.0.0.0:*
udp      0      0 0.0.0.0:3049     0.0.0.0:*
udp      0      0 0.0.0.0:111      0.0.0.0:*
udp      0      0 10.0.0.36:123    0.0.0.0:*
udp      0      0 127.0.0.1:123    0.0.0.0:*
udp      0      0 0.0.0.0:123      0.0.0.0:*
(...)

### ps -auwwwx ###
USER      PID %CPU %MEM    VSZ   RSS TTY  STAT  START   TIME COMMAND
root         1  0.0  0.3  1384   340 ?    S    Jun25   0:31 init [3]
root         2  0.0  0.0     0     0 ?    SW   Jun25   0:00 [keventd]
root         3  0.0  0.0     0     0 ?    SW   Jun25   0:00 [kapm-idled]
root         8  0.0  0.0     0     0 ?    SW   Jun25   0:39 [kupdated]
root        468  0.0  0.6  1452   584 ?    S    Jun25   0:05 syslogd -m 0
root        473  0.0  1.0  2036   980 ?    S    Jun25   0:00 klogd -2
root       32404  0.0  0.0     0     0 ?    Z    Jul11   0:00 [sh <defunct>]
root       32403  0.0  0.0    948    32 ?    T    Jul11   0:00 /bin/sh -c sa1
root        950  0.0  0.7  1580   684 ?    S    Jul12   0:00 CROND
(...)

### vmstat ###
procs  -----memory-----  -swap-  --io--  -system-  --cpu---
r b w  swpd free buff cache  si so  bi bo  in cs  us sy id
0 0 0  1956 3096 1540 37336  0  0  0  0  8 11  0 1  6
(...)
```



Developed Tools

```
### rpcinfo -p 0 ###
  program vers proto  port
    100000    2   tcp   111  portmapper
    100000    2   udp   111  portmapper
    100024    1   udp   1024 status
    100024    1   tcp   1024 status

### uptime ###
12:46pm up 32 days, 0 min,  1 user,  load average: 0.28, 0.42, 0.25

### printenv ###
HOSTNAME=hp1.localdomain
SHELL=/bin/bash
TERM=linux
HISTSIZE=1000
USER=root
(...)
```



Conclusion

- Creation of procedures and tools \Rightarrow great relevance to honeynet researchers.
- √ Automate the processes and standardize information gathering methods.
- √ Avoid human errors.
- √ Reduce the out of operation time of a honeypot.
- √ Importance of developing tools \Rightarrow help with the management difficulties.



Contacts

Carlos Henrique P C Chaves

`carlos.chaves@cenpra.gov.br`

Lucio Henrique Franco

`lucio.franco@cenpra.gov.br`

Antonio Montes

`antonio.montes@cenpra.gov.br`