Nmap

pamN

Kurt Grutzmacher
grutz@jingojango.net

Garrett Gee
ggee@srtek.net
BayLISA – 02/18/06

Who we are

- Penetration testers for a large financial institution in the Bay area
- Many years combined experience in performing assessments, red teaming, exploring vulnerabilities, etc.

Why use attack tools?

Difficult to answer in one slide, but lets try

- Best way to ensure patch application
- The bad guys don’t care about your policy
- Helps gain a deeper knowledge
- Validate protective systems or disprove vendor claims! (Lots of fun!)

Attack Methodology

- Reconnaissance
  - Port map, hunt for Visio documents, hang out at the water cooler, pull user listing from yp, active directory, ldap, etc.
- Attack
  - Exploit systems for profit!
- Control
  - Maintain access without detection

Nmap


```
felix/home/fyodor#nmap -PS22,53,80 -p25 -sV -PE -PM  --packet_trace mail.insecure.org

Starting nmap 3.48 ( http://www.insecure.org/nmap/ ) at 2003-12-03 12:23 PST
SENT (0.0090s) ICMP 63.202.174.201 > 205.217.153.50 Echo request (type=8/code=0) ttl=37 id=41612 iplen=28
SENT (0.0110s) ICMP 63.202.174.201 > 205.217.153.50 Address mask request (type=17/code=0) ttl=54 id=26201 iplen=32
SENT (0.0140s) TCP 63.202.174.201:39786 > 205.217.153.50:22 S ttl=51 id=33935 iplen=40 seq=1127710558 win=4096
SENT (0.0170s) TCP 63.202.174.201:39786 > 205.217.153.50:53 S ttl=40 id=45715 iplen=40 seq=1521975134 win=1024
SENT (0.0190s) TCP 63.202.174.201:39786 > 205.217.153.50:80 S ttl=58 id=53835 iplen=40 seq=1064795998 win=3072
RCVD (0.0220s) ICMP 205.217.153.50 > 63.202.174.201 Echo reply (type=0/code=0) ttl=56 id=12276 iplen=28
SENT (0.3240s) TCP 63.202.174.201:39762 > 205.217.153.50:25 S ttl=40 id=57467 iplen=40 seq=3079198756 win=1024
RCVD (0.3370s) TCP 205.217.153.50:25 > 63.202.174.201:39762 SA ttl=56 id=0 iplen=44 seq=3534696662 win=5840 ack=3534696662
NSOCK (0.3480s) TCP connection requested to 205.217.153.50:25 (IOD #1) EID 8
NSOCK (0.3500s) nsock_loop() started (no timeout). 1 events pending
NSOCK (0.3650s) Callback: CONNECT SUCCESS for EID 8 [205.217.153.50:25]
NSOCK (0.3650s) Read request from IOD #1 [205.217.153.50:25] (timeout: 5000ms) EID 18
NSOCK (0.3940s) Callback: READ SUCCESS for EID 18 [205.217.153.50:25] (27 bytes): 220 core.lnxnet.net ESMTP..
NSOCK (0.3940s) Read request from IOD #1 [205.217.153.50:25] (timeout: 4968ms) EID 26
NSOCK (5.3640s) Callback: READ TIMEOUT for EID 26 [205.217.153.50:25]
NSOCK (5.3640s) Write request for 6 bytes to IOD #1 EID 35 [205.217.153.50:25]: HELP..
NSOCK (5.3640s) Read request from IOD #1 [205.217.153.50:25] (timeout: 5000ms) EID 42
NSOCK (5.3710s) Callback: WRITE SUCCESS for EID 35 [205.217.153.50:25]
NSOCK (5.3860s) Callback: READ SUCCESS for EID 42 [205.217.153.50:25] (55 bytes): 214 qmail home page: http://pobox.com/~djb/qmail.html..
Interesting ports on core.lnxnet.net (205.217.153.50):
PORT   STATE SERVICE VERSION
25/tcp open  smtp    qmail smtpd

Nmap run completed -- 1 IP address (1 host up) scanned in 5.416 seconds
felix/home/fyodor#
```

Portscanning 101

- Objective:
  - Find open TCP and/or UDP listeners on a single or range of TCP/IP Addresses
  - Find out software versions
  - Find out operating system type
  - Don’t get caught doing it
  - Learn what you have on your network

Is Nmap the best tool?

- Yes it is
  - Long history of development and support
  - Active user base, used in many products
  - Continuous development and improvements
  - “Industry Standard” port scanner
- It’s free, open and well documented.
- Stay current! (4.01 as of this doc)

History of Nmap

- First released September 1, 1997 in Phrack 51 “The Art of Portscanning”

  http://www.insecure.org/nmap/p51-11.txt
- Many updates since then:
  - OS Detection (Phrack 54)
  - Idle scanning
  - Version scanning
  - ARP Scanning

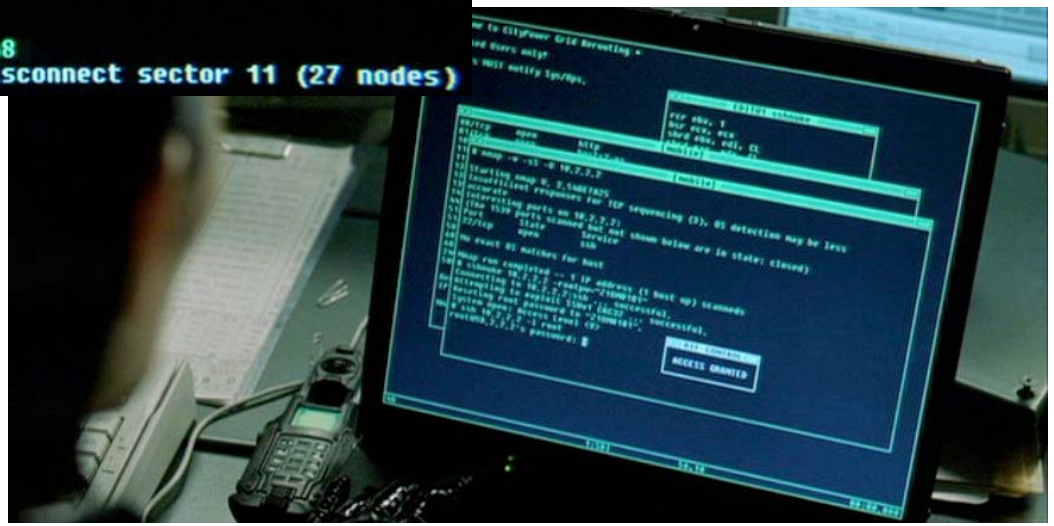
…As seen in the movies!

```
Port        State       Service
22/tcp      open        ssh

No exact OS matches for host

Nmap run completed -- 1 IP address (1 host up) scanneds
# sshnuke 10.2.2.2 -rootpw="Z1ON0101"
Connecting to 10.2.2.2:ssh ... successful.
Attempting to exploit SSHv1 CRC32 ... successful.
Reseting root password to "Z1ON0101".
System open: Access Level <9>
# ssh 10.2.2.2 -l root
root@10.2.2.2's password:

RRF-CONTROL> disable grid nodes 21 - 48
Warning: Disabling nodes 21-48 will disconnect sector 11 (27 nodes)
```



Host Discovery

- TCP SYN Probe (-PS<portlist>)
- TCP ACK Probe (-PA<portlist>)
- UDP Probe (-PU<portlist>)
- ICMP Echo Request/Ping (-PE)
- ICMP Timestamp Requset (-PP)
- ICMP Netmask Request (-PM)
- ARP Probes (-PR)

Most valuable TCP ‘ping’ Ports?

- 80 (http)
- 25 (smtp)
- 22 (ssh)
- 443 (https)
- 21 (ftp)
- 113 (auth)
- 23 (telnet)
- 53 (domain)
- 554 (rtsp)
- 3389 (ms-term-server)
- 1723 (pptp)

TCP SYN or ACK Probes?

- Send both!
  - Purpose is to find hosts that are up
  - We do not care whether the port is active yet

Most valuable UDP “Ping” Port

- Pick a high numbered one
  - Anything that responds with ICMP is up
  - Most things respond with ICMP
- UDP “Ping” scanning is very… Um.. What’s that word… Ohhh yeah:

UNRELIABLE

Most Valuable ICMP “Ping” Types

- Echo Request (-PE)

  …plus either Timestamp (-PP)

  …or Netmask (-PM)

ARP Ping Probing

- Useful only on same subnet
- VERY reliable and much faster
- Sends raw ethernet ARP requests
- Automatically used if host/network is on the local subnet
  - Unless --send-ip option specified

Intense Discovery!

```
# nmap –sP –PE –PP –PS21,22,23,25,80,113,21339
      –PA80,113,443,10042 –source-port 53 –n
      –T4 –iR 10000
```

[… lots of IPs …]

```
Host a.b.c.d appears to be up.
Host w.x.y.z appears to be up.
Nmap finished: 10000 IP addresses (699 hosts up)
  scanned in 2016.564 seconds
```

Mission!

Locate webserver(s) on the Playboy.com network that may be offering free images

(yeah, overused but it’s effective)

Step 1

Find network(s) to scan:

$ whois -h whois.arin.net n playboy

Playboy PLAYBOY-BLK-1 (NET-216-163-128-0-1)
216.163.128.0 - 216.163.143.255

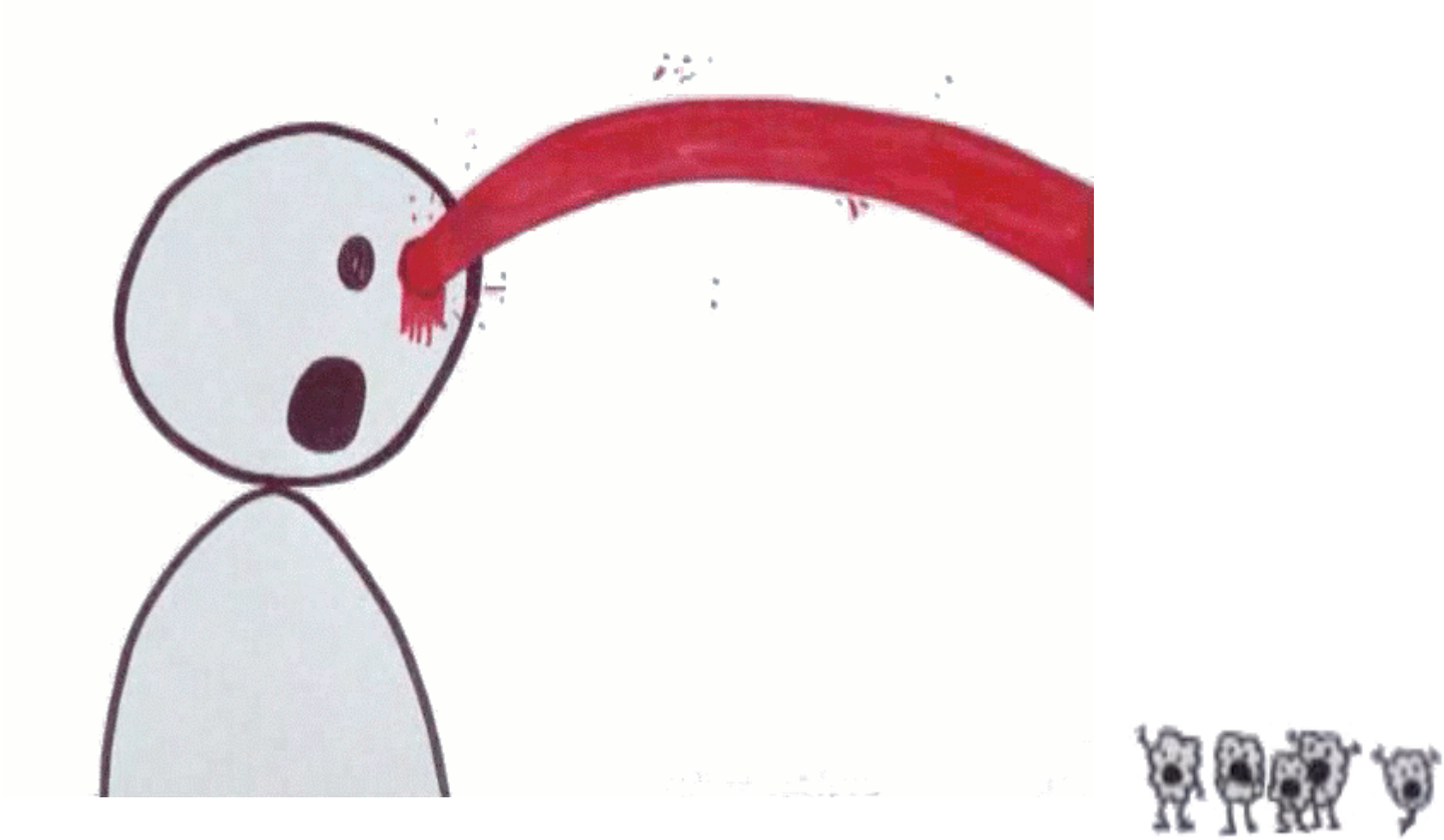
PLAYBOY ENTERPRISES INC-050926150640
SBC071137177168290509261506 44 (NET-71-137-177-168-1) ***71.137.177.168 - 71.137.177.175***

First Try (the kiddie way)

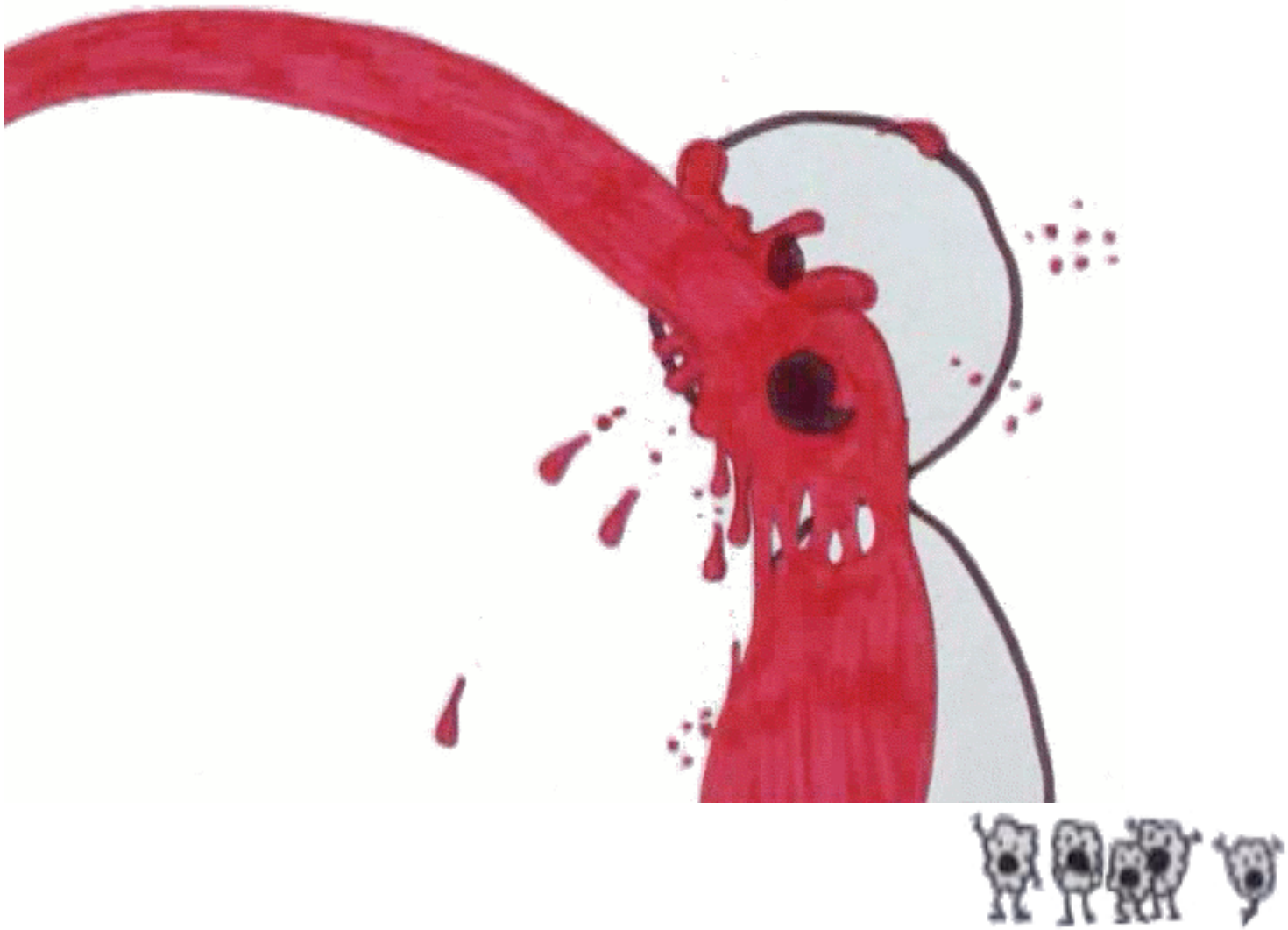
```
# nmap -P0 -p80 -oG pb.gnmap 216.163.128.0/20
[...]
Nmap finished: 4096 IP addresses (4096 hosts up)
   scanned in 457.658 seconds

$ egrep '[^0-9]80/open' pb.gnmap  | wc
     35     175    2424
```

What's the key to comedy?



TIMING!



Timing

- host www.playboy.com

  www.playboy.com has address 216.163.137.3
- host –t mx playboy.com

  mx.chi.playboy.com (216.163.143.4)

  mx.la.playboy.com (216.163.128.15)
- Try to ping?
  - No response… Damn.

TCP Ping?

```
hping2 --syn -p 25 -c 5 -L 0 mx.la.playboy.com
HPING mx.la.playboy.com (eth0 216.163.128.15): S set, 40 headers + 0 data bytes
len=46 ip=216.163.128.15 ttl=49 DF sport=25 flags=SA seq=0 win=65535 rtt=32.3 ms
[ … ]
--- mx.la.playboy.com hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 31.4/39.0/53.7 ms

hping2 --syn -p 25 -c 5 -L 0 mx.chi.playboy.com
HPING mx.chi.playboy.com (eth0 216.163.143.4): S set, 40 headers + 0 data bytes
len=46 ip=216.163.143.4 ttl=50 DF sport=25 flags=SA seq=0 win=65535 rtt=59.6 ms
[ … ]
--- mx.chi.playboy.com hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 58.7/59.0/59.6 ms
```

Lets Try That Again

```
# nmap -T4 --max-rtt-timeout 200 --initial-rtt-
   timeout 150 --min-hostgroup 512 -P0 -p80 -oG
   pb2.gnmap 216.163.128.0/20
```

Nmap finished: 4096 IP addresses (4096 hosts up) scanned in *230.118* seconds (previous was *457.658* seconds!)

```
$ egrep '[^0-9]80/open' pb2.gnmap  | wc
     35     175    2424
```

Can we do it faster?

Remove DNS (add the –n option)

Nmap finished: 4096 IP addresses (4096 hosts up) scanned in *163.725* seconds

```
$ egrep '[^0-9]80/open' pb3.gnmap  | wc
     35     175    1814
```

Second Mission

Find open TCP ports on www.baylisa.org

```
# nmap -sS -T4 www.baylisa.org

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )

Warning: Finishing early because retransmission cap hit.
Interesting ports on www.baylisa.org (205.217.155.154):
(The 1667 ports scanned but not shown below are in state: closed)
PORT     STATE    SERVICE
22/tcp   open     ssh
25/tcp   open     smtp
80/tcp   open     http
1453/tcp filtered genie-lm
3306/tcp open     mysql

Nmap finished: 1 IP address (1 host up) scanned in 45.720 seconds
```

FIN Scan

```
# nmap -sF -T4 www.baylisa.org

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )
All 1672 scanned ports on www.baylisa.org
   (205.217.155.154) are: open|filtered

Nmap finished: 1 IP address (1 host up) scanned in 35.419
   seconds
```

ACK Scan

```
# nmap -sA -T4 www.baylisa.org

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )
All 1672 scanned ports on www.baylisa.org
  (205.217.155.154) are: UNfiltered

Nmap finished: 1 IP address (1 host up) scanned in
  46.347 seconds
```

Window Scan

- Explain?
  - RST with Zero Window – Port Closed
  - RST with Positive Window – Port Open

Window Scan

```
# nmap -sW -T4 www.baylisa.org

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )
All 1672 scanned ports on www.baylisa.org
  (205.217.155.154) are: closed

Nmap finished: 1 IP address (1 host up) scanned in
  46.866 seconds
```

Version Scanning

```
# nmap -sV -T4 www.baylisa.org

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )

Interesting ports on www.baylisa.org (205.217.155.154):
(The 1668 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     SunSSH 1.0.1 (protocol 2.0)
25/tcp   open  smtp    Sendmail 8.13.1
80/tcp   open  http    Apache httpd 1.3.29 ((Unix) PHP/4.4.0 mod_perl/1.25)
3306/tcp open  mysql   MySQL (unauthorized)
Service Info: OS: Unix

Nmap finished: 1 IP address (1 host up) scanned in 55.499 seconds
```

Enterprise Scanning?

- Some things that may help:
  - XML Output
  - Nmap::Parser Perl Module
  - Nmap::Scanner Perl Module
  - More at http://www.insecure.org/nmap

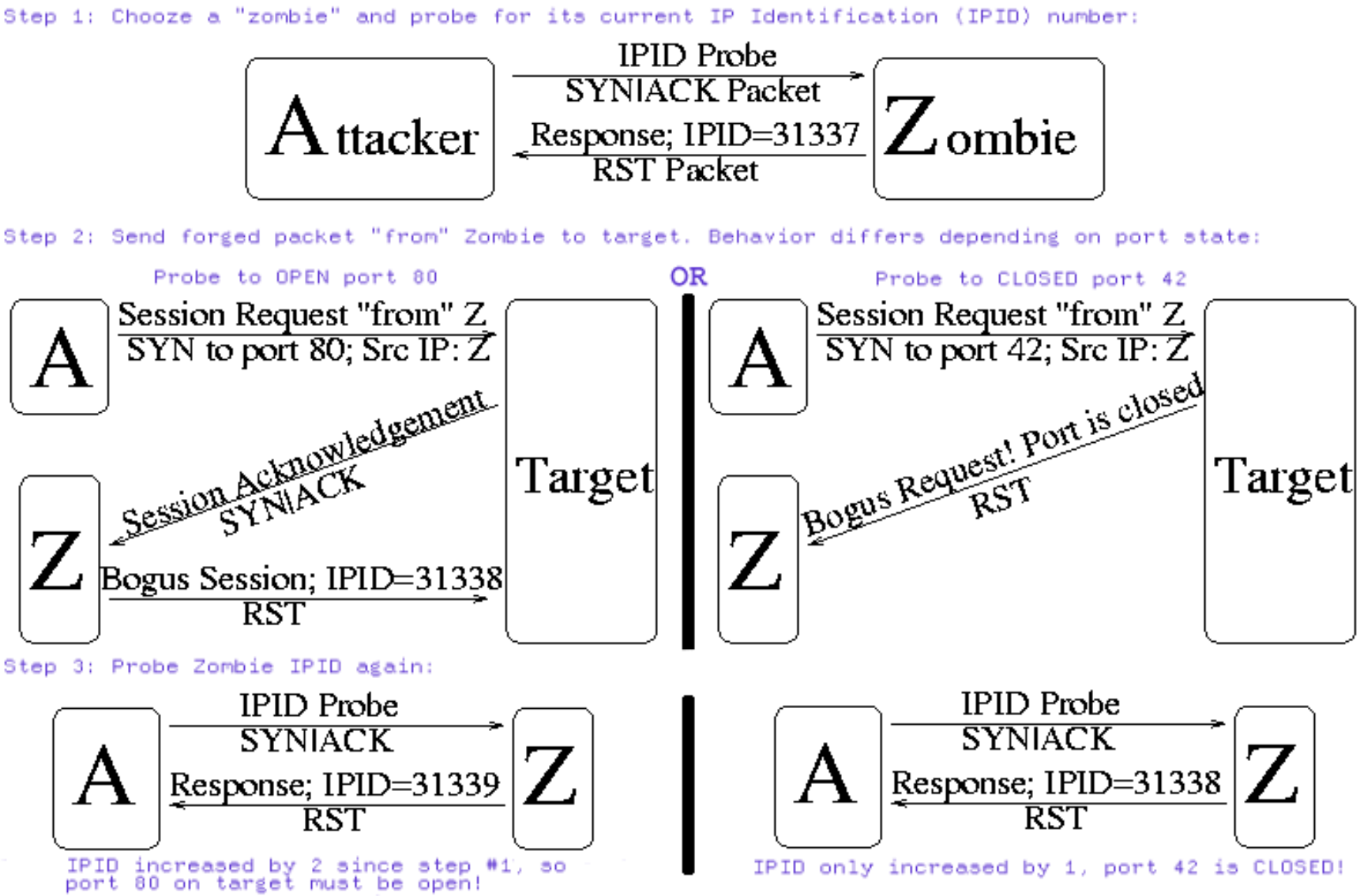
Other Things Nmap does

- RPC Scanning (rpcinfo –p <host>)
- FTP Bounce scanning
- Working on scanning from SOCKS/HTTP Proxies
- Idle Host Scanning (-sI <host:port>)
- Protocol scanning (-sO)
- Very definable options for timing, TTL, data length, max retries, max host groups, bogus TCP/UDP checksums, IPv6, raw ethernet or IP frames, etc.
- nmap –h

Idle Scanning

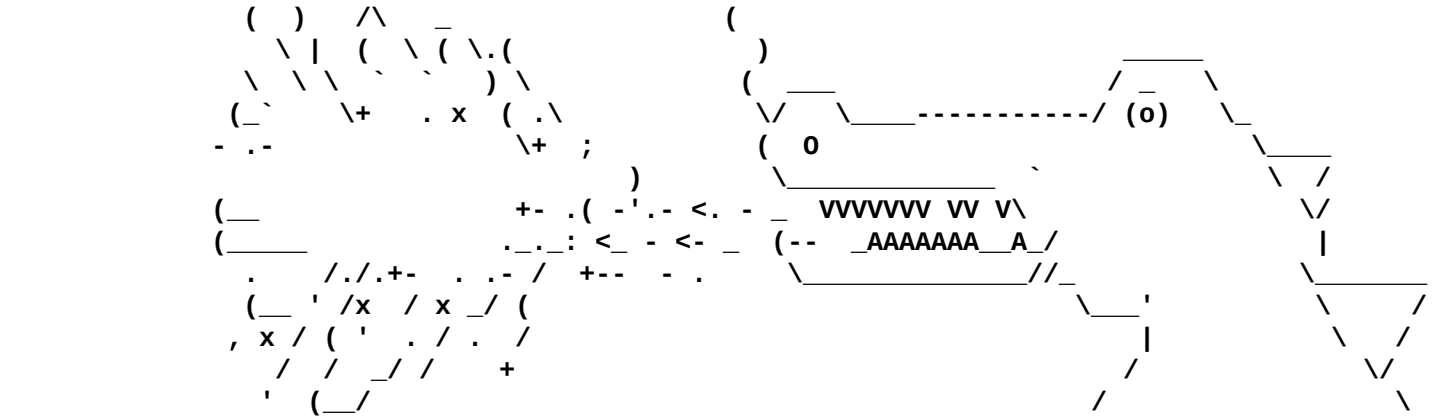
- Uses IPID on a quiet host (zombie) to check for open ports on other hosts.
- Nmap spoofs as the zombie sending packets to the victim
- Polls the zombie to see if IPID has incremented (received RST from victim when not expecting)

Nmap Idle Scan Technique (Simplified)
http://www.insecure.org



Nmap on I(ntel)Mac!

```
grutzImac:~/nmap grutz$ uname -a
Darwin grutzImac.local 8.4.1 Darwin Kernel Version 8.4.1: Tue Jan  3 18:23:53 PST 2006; root:xnu-
    792.7.90.obj~1/RELEASE_I386 i386 i386
grutzImac:~/nmap grutz$ tar -xjf nmap-4.00.tar.bz2
grutzImac:~/nmap grutz$ cd nmap-4.00
grutzImac:~/nmap/nmap-4.00 grutz$ ./congfigure --with-libpcre=/opt/local    ← use another pcre so far
[ ... ]
checking build system type... i386-apple-darwin8.4.1
checking host system type... i386-apple-darwin8.4.1
configure: creating ./config.status
config.status: creating Makefile
```



```
            NMAP IS A POWERFUL TOOL -- USE CAREFULLY AND REPONSIBLY
Configuration complete.  Type make (or gmake on some *BSD machines) to compile.
```

Nmap on I(ntel)Mac!

```
grutzImac:~/nmap/nmap-4.00 grutz$ make
Compiling libdnet
[ ... ]
grutzImac:~/nmap/nmap-4.00 grutz$ ./nmap -h
Nmap 4.00 ( http://www.insecure.org/nmap/ )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sP: Ping Scan - go no further than determining if host is online
  -P0: Treat all hosts as online -- skip host discovery
  -PS/PA/PU [portlist]: TCP SYN/ACK or UDP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idlescan
  -sO: IP protocol scan
  -b <ftp relay host>: FTP bounce scan
```

Nmap on I(ntel)Mac!

```
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080
  -F: Fast - Scan only the ports listed in the nmap-services file)
  -r: Scan ports consecutively - don't randomize
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
OS DETECTION:
  -O: Enable OS detection
  --osscan-limit: Limit OS detection to promising targets
  --osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
  -T[0-5]: Set timing template (higher is faster)
  --min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
  --min-parallelism/max-parallelism <msec>: Probe parallelization
  --min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <msec>: Specifies
      probe round trip time.
  --max-retries <tries>: Caps number of port scan probe retransmissions.
  --host-timeout <msec>: Give up on target after this long
  --scan-delay/--max-scan-delay <msec>: Adjust delay between probes
```

Nmap on I(ntel)Mac!

```
FIREWALL/IDS EVASION AND SPOOFING:
  -f; --mtu <val>: fragment packets (optionally w/given MTU)
  -D <decoy1,decoy2[,ME],...>: Cloak a scan with decoys
  -S <IP_Address>: Spoof source address
  -e <iface>: Use specified interface
  -g/--source-port <portnum>: Use given port number
  --data-length <num>: Append random data to sent packets
  --ttl <val>: Set IP time-to-live field
  --spoof-mac <mac address/prefix/vendor name>: Spoof your MAC address
  --badsum: Send packets with a bogus TCP/UDP checksum
OUTPUT:
  -oN/-oX/-oS/-oG <file>: Output scan in normal, XML, s|<rIpt kIddi3,
     and Grepable format, respectively, to the given filename.
  -oA <basename>: Output in the three major formats at once
  -v: Increase verbosity level (use twice for more effect)
  -d[level]: Set or increase debugging level (Up to 9 is meaningful)
  --packet-trace: Show all packets sent and received
  --iflist: Print host interfaces and routes (for debugging)
  --append-output: Append to rather than clobber specified output files
  --resume <filename>: Resume an aborted scan
  --stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML
  --webxml: Reference stylesheet from Insecure.Org for more portable XML
  --no-stylesheet: Prevent associating of XSL stylesheet w/XML output
```

Nmap on I(ntel)Mac!

```
MISC:
  -6: Enable IPv6 scanning
  -A: Enables OS detection and Version detection
  --datadir <dirname>: Specify custom Nmap data file location
  --send-eth/--send-ip: Send using raw ethernet frames or IP
   packets
  --privileged: Assume that the user is fully privileged
  -V: Print version number
  -h: Print this help summary page.
EXAMPLES:
  nmap -v -A scanme.nmap.org
  nmap -v -sP 192.168.0.0/16 10.0.0.0/8
  nmap -v -iR 10000 -P0 -p 80
SEE THE MAN PAGE FOR MANY MORE OPTIONS, DESCRIPTIONS, AND
   EXAMPLES
```

Nmap on I(ntel)Mac!

```
grutzImac:~/nmap/nmap-4.00 grutz$ sudo ./nmap -sV www.microsoft.com
Password:

Starting Nmap 4.00 ( http://www.insecure.org/nmap/ )
Verbosity Increased to 1.
Verbosity Increased to 2.
The SYN Stealth Scan took 20.90s to scan 1672 total ports.
Initiating service scan against 2 services on 207.46.20.60 at 11:39
The service scan took 17.39s to scan 2 services on 1 host.
Host 207.46.20.60 appears to be up ... good.
Interesting ports on 207.46.20.60:
(The 1670 ports scanned but not shown below are in state: filtered)
PORT    STATE SERVICE  VERSION
80/tcp  open  http     Microsoft IIS webserver 6.0
443/tcp open  ssl/http Microsoft IIS webserver 6.0
Service Info: OS: Windows

Nmap finished: 1 IP address (1 host up) scanned in 38.588 seconds
               Raw packets sent: 3346 (134KB) | Rcvd: 7 (332B)
```

Questions?

Thanks!

- Thanks to Fyodor for writing a great tool and offering us this gig.

- http://grutz.jingojango.net/presentations/