Wireless Device Fingerprinting: Techniques and Application

Joshua Wright
Aruba Networks
jwright@arubanetworks.com
401-524-2911

ARUBA™
The Mobile Edge Company

Introduction

- Assessment of Techniques
- "Bad" uses of fingerprinting
- "Good" uses of fingerprinting
- Fingerprinting impact on WIDS

Assessment of Techniques

- Recent papers cite new techniques to fingerprint wireless devices
- Techniques support varying levels of granularity for analysis
- WIDS countermeasures fingerprinting
- Three predominant client-identification techniques
  - Response to malformed stimuli analysis; active
  - Probe request timing deltas; passive
  - Duration and frame type analysis; passive

WIDS Countermeasures

- Many WIDS vendors apply "countermeasures" against rogue devices
  - A.k.a WIPS, RF Shield, <insert marketing here>
  - Typically deauth/dissasoc floods to repeatedly disconnect STA
- Different vendors use different countermeasure techniques
  - Of 4 major overlay vendors, each is unique enough to identify the WIDS implementation
- Status: Manual analysis (Wireshark) required, fingerprints need refresh with updated products

WIDS Fingerprinting Inputs

- Disconnect technique: deauth, disassociate, both?
- Disconnect direction: Message to client, message to AP, both?
- Reason code, AP → STA
- Reason code, STA → AP
- Timing between disconnect messages
- Sequence number selection of spoofed disconnect frames
- Other vendor-unique attributes

Vendor WIPS Implementations

Vendor	Version Tested	Analysis[1]
Network Chemistry	RF Protect 4.05	Deauth sent bidirectionally, AP→STA uses reason code 2, STA → AP uses reason code 3, fixed timing .10s, sequential seq# starting at 0, sequential frag# starting at 0
AirTight	SpectraGuard 3.0.08	Deauth sent bidirectionally, AP→STA uses reason code 2, STA → AP uses reason code 3, fixed timing .15s, seq# fixed at 0 for STA → AP, seq# fixed at 1 for AP → STA
AirDefense	AirDefense Enterprise 6.0	Combined deauth and disassociate, AP→STA messages only, reason code #2 in both deauth and disassociate, intelligent timing variable based on response from target, sequential seq# starting at 0
AirMagnet	AirMagnet Enterprise 5.2.0	Deauth sent bidirectionally, AP→STA uses reason code 5, STA → AP uses reason code 3, timing with first frame at .03 second delay followed by 5 frames at .02 second delay, seq# fixed at 0 for both STA → AP and AP → STA

[1] Analysis completed May 2005, http://i.cmpnet.com/nc/1612/graphics/SessionContainment_file.pdf

Why is WIDS Fingerprinting Useful?

Attacks Detected and Classified

	Publicly released	Developed by reviewer	Provided by vendor	Detected attack AT	Detected attack AD	Detected attack AM	Detected attack NC	Accurately classified AT	Accurately classified AD	Accurately classified AM	Accurately classified NC
Deauthenticate flood	Y	Y	N	●	●	●	●	●	●	●	●
Disassociate flood	Y	Y	N	●	●	●	●	●	●	●	●
Deauthenticate broadcast flood	Y	Y	N	●	●						
Disassociate broadcast flood	Y	Y	N	●	●						
Clear-to-send flood	Y	Y	N	●	○						
Request-to-send flood	Y	Y	N	●	●						
EAPOL logoff flood	N	N	Y	●	○						
EAP failure flood	N	N	Y	○	○						
NULL SSID DoS	N	Y	N	●	●						
Unknown authentication algorithm	Y	N	N	●	●						
DSSS test mode jamming	N	Y	N	○	○						
PEAP authentication failure	Y	N	N	○	●						
Authentication flood	N	Y	N	●	●	●	●	●	○	●	○
Combined authenticate/associate flood	N	Y	N	●	●	●	●	●	○	●	○
IDS evasion, EAPOL logoff flood	Y	Y	N	○	○	○	●				●
IDS evasion, EAP failure flood	Y	Y	N	○	○	○	●				●
Session hijacking	N	Y	N	○	○	○	○				
ICV invalidation for plain-text recovery	Y	N	N	●	●	○	●	○	○		○
Too large fragment	N	Y	N	○	●	●	●		○	○	○
Incomplete fragment flood	N	Y	N	○	●	●	●		○	○	○
Out-of-order fragments	N	Y	N	○	●	●	●		○	○	○
Initial small, then large fragments	N	Y	N	○	●	●	●		○	○	○
Evil twin attack	Y	N	N	●	●	●	●	●	○	●	●
Fake AP advertisement flood	Y	N	N	○	●	●	●		●	○	○
Active network scanning	Y	N	N	○	○	●	●			○	○
Fragmented packet injection	N	Y	N	○	○	○	○				
Traffic replay	Y	N	N	●	●	○	●	○	○		●
MAC spoofing	Y	N	N	○	●	○	●		●		●
PRGA packet crafting and injection	Y	N	N	○	●	●	●		○	○	●

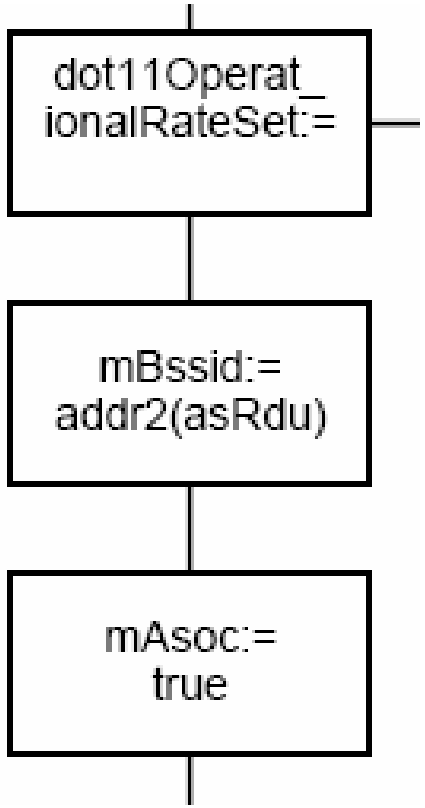
Y, ●=Yes N, ○=No AT=AirTight AD=AirDefense AM=AirMagnet NC=Network Chemistry

Sad truth of WIDS systems: The best vendors detect less than 60% of attacks.

ARUBA™
The Mobile Edge Company

Malformed Stimuli Response

- Concept: Observe how clients react to malformed stimuli
  - Very similar to nmap approach
- Classifies STA driver, chipset, OS
- Malformed frame techniques:
  - Assoc. Resp. exchanges SRC for BSSID
  - Unprovisioned frame types
  - Unprovisioned reason codes in deauth, disassociate frames
- Some stations drop frames
- Others send deauth/dissasoc, unique reason codes



802.11-1999, pg 376

Malformed Stimuli Response - Analysis

- Advantages of this technique:
  - Quick to assess station; barrage of frames transmitted quickly, identification follows
  - Empirical analysis suggests strong accuracy levels
- Disadvantages of this technique:
  - Potential to disrupt the network
  - WIDS perspective: Can be detected
  - Low reporting fidelity, cannot differentiate between driver revisions in most cases
  - Station must be associated
- Status: Manually implemented with test cases, not publicly available

Probe Response Timing

- Concept: Use timing delta and frequency of probe request frames to fingerprint
  - "Sandia Technique"
- Classifies STA driver, chipset, OS, mgmt utility
- Analysis uses "binning" technique, supervised Bayesian classification approach
  - Multiple bins are created, each representing delta of .8 second intervals from previous probe request
  - Each bin records percentage of frames for the bin, mean time delta within this bin

Probe Response Timing - Fingerprints

engenius-unassoc-win (0,0.676,0.16)(1,0.228,1.42)(50,0.96,39.8)

Bin	Percentage	Mean	Note
0	.676	.16	More than 67% of all probe request frames were transmitted within 0 - 0.8 seconds with a mean delta of .16 seconds
1	.228	1.42	More than 22% of all probe request frames were transmitted within 0.8 and 1.6 seconds with a mean delta of 1.42 seconds
50	.096	39.8	A very small number of probe request frames were transmitted with a delta mean of 39.8 seconds

- Using this technique, authors claim to identify:
  - STA card type (e.g. cisco-abg), and by association, card chipset
- Windows WLAN manager, WZC or other
- Limited driver version information

Probe Response Analysis - Accuracy

- Authors evaluated their technique with 17 different wireless card/driver/management-tool implementations
  - 9 drivers were fingerprinted with 100% accuracy
  - 3 drivers were fingerprinted with 99%-90% accuracy
  - 2 drivers were fingerprinted with 89%-80% accuracy
  - 1 drivers were fingerprinted with 79%-70% accuracy
  - 2 drivers were fingerprinted with 69%-60% accuracy
- Status: Fingerprints of test cases released publicly, no implementations have been released

Probe Response Timing - Analysis

- Advantages of this technique:
  - Passive analysis is desirable, avoids detection, no potential to disrupt network
  - Can be applied to stations regardless of association
  - Can differentiate WZC vs. third-party control
- Disadvantages of this technique:
  - Low reporting fidelity, cannot differentiate between driver revisions in most cases
  - Accuracy is questionable for some devices
- Analysis is flawed however, testing with a single SSID in the PNL
  - Not a realistic test environment, results suspect

Dur. and Frame Type Fingerprinting

- Concept: Use duration values and frame types to classify
  - "Ellch Technique", Jon Ellch thesis
- Classifies STA driver, chipset, OS
- Analysis uses ratio of duration compared to total for each unique duration
  - Applied to all frames regardless of type/subtype, and to each unique type/subtype

Dur. and Frame Type - Calculation

Analysis for a single duration value, regardless of frame type:

```
duration_ratio(d) = # of packets with duration d / # of total packets
sum=0
for every duration-value d inclusive (observed packets | fingerprint)
  sum += 1.0 - | L.duration_ratio(d) - R.duration_ratio(d)|
return sum
```

Analysis for duration values and frame type information:

```
duration_ratio(p,d) = # of packets with packet type p and duration d /
# of packets with packet type p
sum=0
for every pair(packet type p, duration value d) inclusive (observed
packets | fingerprint)
  sum += 1.0 - | L.duration_ratio(p,d) - R.duration_ratio(p,d)|
return sum
```

Results of two calculations are added for a final score

L is the observed packet capture, R is the fingerprint data

ARUBA™
The Mobile Edge Company

Dur. and Frame Type - Caveat

- Duration values naturally change depending on network characteristics, including:
  - Short or long preamble in use
  - Short Slot Time (SST) enabled or disabled
  - Modulation DSSS/OFDM (b/g, 802.11a not yet assessed)
- As a result, fingerprints need to be generated for each of these parameters
  - e.g. Each driver/card combination requires multiple fingerprint depending on network

Dur. and Frame Type - Fingerprint

Fingerprint: Atheros 5211, Windows XP driver version 3.3.0.1561

Packet Type	(duration [duration observed frequency/#packet of this type])	Note
Associate Request	(314 [2/2])	2 assoc. request frames out of 2 total had a duration value of 314
Probe Request	(0 [75/77]) (314 [2/77])	75 out of 77 probe request frames had a duration value of 0, the remaining 2 had a duration value of 314
Authenticate Request	(314 [2/2])	2 out of 2 authentication request frames had a duration value of 314
Data	(162 [167/278]) (0 [111/278])	167 out of 278 data frames have a duration value of 162, while 111 out of 278 have a duration value of 0)
NULL Data	(162[597/597])	100% of NULL data frames have a duration of 167

Dur. and Frame Type - Accuracy

- Author used 14 cards and driver versions for analysis, reports over 97% accuracy
  - Able to differentiate between versions of driver software as well
  - Also able to differentiate between two different chips (Atheros 5212, 5211) supported by same driver
- Private implementation shared upon request

Dur. and Frame Type - Example

```
-----Loaded 13 prints------
0 47.5884 3  Proxim-Orinoco Silver AR5211    Windows XP-SP2 ntpr11ag.sys-3.1.2.219
1 47.1830 2  Proxim-Orinoco Silver AR5212    Windows XP-SP2 ntpr11ag.sys-3.1.2.219
2 45.7977 4  Proxim-Orinoco Silver AR5212    Windows XP-SP2 ntpr11ag.sys-3.1.2.219
3 45.6490 1  Linksys-WPC55AG       AR5212    Windows XP-SP2 ar5211.sys-3.3.0.1561
4 30.2763 7  Intel-IPW220BG        IPW2200BG Windows XP-SP2 w29n51.sys-90039
5 30.1016 10 Broadcom-MiniPCI      BCM4318   Windows XP-SP2 bcmwl5.sys-3.100.46.0
The closest match I have in the DB is: implementation #3
Vendor: Proxim
Model:  Orinoco Silver
Version:        8461-05
OS:             Windows XP
OS-Version:     SP2
chipset-vendor: Atheros
chipset:        AR5211
driver-name:    ntpr11ag.sys
driver-version: 3.1.2.219
```

ARUBA™
The Mobile Edge Company

Dur. and Frame Type - Analysis

- Advantages of this technique:
  - Most detailed granularity in analysis
  - Strong result accuracy
  - Results improve in accuracy as 802.11 develops over time (e.g. new types added with 802.11e/k/r/w, new PHY added with 802.11y)
- Disadvantages of this technique:
  - Station must be associated to fingerprint
  - Analysis must observe multiple frame types (most effective when observations start at probe req.)
  - Complex signature mechanism; how to handle roaming and traffic across varying AP's/SST/modulation/rates, etc.
  - No client manager identification

Additional Analysis Techniques

- OUI matching
- PHY type exclusion
- Listen interval analysis in authentication request frames
  - Empirical analysis: 93 random clients, 67% have LI of 0x01, 30% have LI of 0xc8
- Supported encryption mechanisms
- Power save behavior characteristics
- AP search algorithm
  - Probe request channel sequences
  - Known BSSID caching or repeated probe before association
  - Use of broadcast vs. directed probe requests
- Association characteristics
  - Does STA send deauth old association, or just abandon and timeout?
  - Does client use reassociation type, or just associate?

"Bad" Uses of Fingerprinting

- Target analysis and network mapping
  - Passive monitoring of encrypted traffic reveals OS in use and driver versions
  - Can disclose patching practices, and by association the organization security posture
- Driver exploit targeting
  - Exploits are often OS and driver version specific
  - For success, attacker must fingerprint target to select appropriate exploit RET address

"Good" Uses of Fingerprinting

- Vulnerability assessment of organization
  - Identify all driver versions
  - Compare to known database of vulnerabilities
- Assists in location analysis using round-trip calculations
  - RT analysis depends on chipset in use
- Client troubleshooting
  - Driver issues often lead to connectivity problems
  - Identification of drivers can help troubleshoot and improve client connectivity
- Clientless third-party patch management reporting
- Client remediation; only allow up-to-date drivers to access the network, quarantine the rest (NAC approach)

WIDS Impact

- One emerging wireless attack is to exploit driver flaws
  - Buffer/heap overflow exploits are starting to appear that target 802.11 driver code
- Some exploits can be detected with layer 2 WIDS analysis
  - Broadcom target, malformed probe response
- Other exploits cannot be detected with only layer 1/layer 2 analysis

Driver Vulnerability Disclosure

Triggering the race condition is fairly easy.

1) set up a netcat udp listener on the victim
2) start blasting udp packets at it from a machine. sleeping for about 4000 microseconds between packets seems to be a good start.
3) start flooding the victim machine with disassociation requests. A BSOD should follow very shortly. A delay of 5000 microseconds between packets seems useful.

If you're lucky, your UDP packet will end up on the stack. If you're less lucky, a beacon packet from a nearby network will end up on the stack.

Subject: "Re: [Dailydave] This guy cracks me up. (MindsX)"
http://archives.neohapsis.com/archives/dailydave/2006-q3/0184.html

ARUBA™
The Mobile Edge Company

Windows Driver Crash-Dump

```
kd> !analyze -v

DRIVER_IRQL_NOT_LESS_OR_EQUAL (d1)
An attempt was made to access a pageable (or completely invalid) address at an
interrupt request level (IRQL) that is too high.  This is usually
caused by drivers using improper addresses.
If kernel debugger is available get stack backtrace.
Arguments:
Arg1: 5c01abf7, memory referenced
Arg2: 00000002, IRQL
Arg3: 00000001, value 0 = read operation, 1 = write operation
Arg4: cccccccf, address which referenced memory

Debugging Details:
------------------
WRITE_ADDRESS:  5c01abf7
CURRENT_IRQL:  2
FAULTING_IP:
+ffffffffcccccccf
cccccccf 01963b10ffd6     add     [esi+0xd6ff103b],edx
^-----payload of UDP packet in EIP. Pwned.
```

Attacker control of Extended Instruction Pointer (EIP) indicates a vulnerability that can be exploited to run arbitrary code

ARUBA™
The Mobile Edge Company

Disparate Analysis Mechanisms

- WIDS System observation:
  - Deauth flood against target; classified as DoS attack
- NIDS (wired) observation:
  - UDP flood against a target, rate anomaly; impact unknown
- Correlation necessary to evaluate true impact
- Not currently available in any products

Leveraging Fingerprinting with WIDS

- Not particularly useful for attacker identification
  - Easy for an attacker to fool fingerprinting system into incorrect characterization
  - PHY-layer fingerprinting technique has more promise for spoofing detection
- Wireless driver vulnerability assessment
  - WIDS system fingerprints observed clients
  - Can generate report of vulnerable stations with simple SQL query
- Prioritization of events
  - Disassociate flood against target identified as vulnerable Centrino NIC; elevate impact to driver exploit

Issues

- Generating fingerprint databases
  - Very difficult to privately develop a comprehensive database of fingerprints
  - Requires "one of each", consider wireless EKG machines; far too costly
  - Possible solution: Integration with Kismet, simple submission system (similar to Nmap approach)
- Duration/type analysis techniques have not been extended to fingerprint 802.11a networks

Summary

- Device fingerprinting has significant potential for damaging and helping WLAN security
- Techniques for identifying WIDS systems
- STA fingerprinting
  - Active with malformed frames
  - Passive with probe request frequency analysis, duration/type analysis
- Fingerprinting improves quality of WIDS reporting, vulnerability assessment