Attacking Automatic Wireless Network Selection

Dino A. Dai Zovi and Shane A. Macaulay
{ddaizovi,smacaulay1}@bloomberg.com

We made Slashdot!

Hackers, Meet Microsoft

"The random chatter of several hundred Microsoft engineers filled the cavernous executive briefing center recently at the company's sprawling campus outside Seattle. Within minutes after their meeting was convened, however, the hall became hushed. Hackers had successfully lured a Windows laptop onto a malicious wireless network. 'It was just silent,' said Stephen Toulouse, a program manager in Microsoft's security unit. 'You couldn't hear anybody breathe.' The demo was part of an extraordinary two days in which outsiders were invited into the heart of the Windows empire for the express purpose of exploiting flaws in Microsoft computing systems. The event, which Microsoft has not publicized, was dubbed 'Blue Hat' -- a reference to the widely known 'Black Hat' security conference, tweaked to reflect Microsoft's corporate color."

Agenda

- Motivation
- What is Automatic Wireless Network Selection?
- Windows XP Wireless Auto Configuration (WZCSVC) Algorithm
- Wireless Auto Configuration Weaknesses and Vulnerabilities
- KARMA: Wireless Client Attack Assessment Toolkit

Motivation

- Wireless LANs now can be and increasingly are quite secure
  - Improved encryption systems (WPA)
  - MAC address filtering
  - Hidden networks (SSID cloaking)
- Mobile clients bridge networks across time
  - Connect to secure networks as well as insecure networks (conferences, hotels, airports, cafes)
  - Can be compromised on airplane and spread compromise to secure network at work
  - Security of most secure network depends on security of least secure network

Motivation

- Paradigm shift to new wireless threat
  - Attacking wireless clients
- Nightmare scenario
  - Target: Identify wireless clients
  - Position: Get on same network as victim
  - Attack: Exploit client-side vulnerabilities to install persistent agent
  - Subvert: Agent gives attacker remote access to secure networks that client connects to

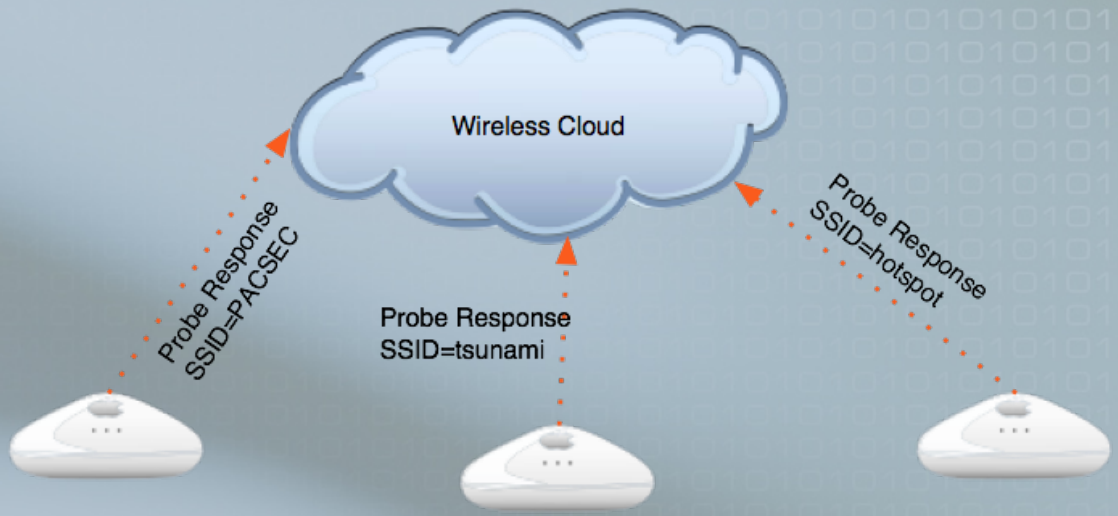
Automatic Wireless Network Selection

- Purpose: Automatically (re)connect to trusted known wireless networks
- Operating System maintains list of *Trusted/ Preferred* wireless Networks
  - Records (SSID, Cleartext/WEP/WPA)
- Preferred Networks are automatically connected to when available
  - Windows: Continually search when wireless card is on and not associated to another wireless network
  - MacOS X: Search for networks when user logs in or machine wakes from sleep

Microsoft Windows XP Wireless Auto Configuration Algorithm

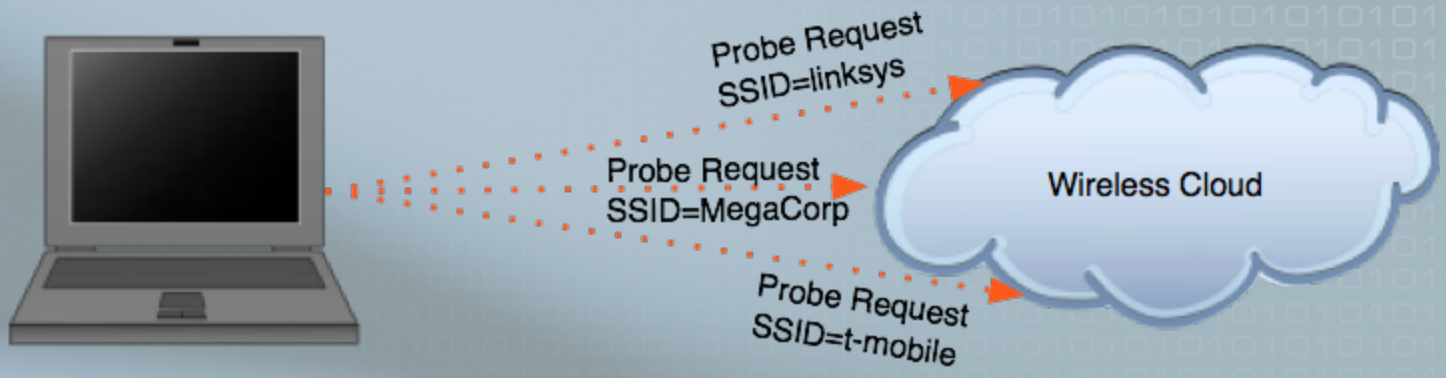
- First, Client builds list of available networks
  - Send broadcast Probe Request on each channel

Wireless Auto Configuration Algorithm



- Access Points within range respond with Probe Responses

Wireless Auto Configuration Algorithm



- If Probe Responses are received for networks in preferred networks list:
  - Connect to them in preferred networks list order
- Otherwise, if no available networks match preferred networks:
  - Specific Probe Requests are sent for each preferred network in case networks are “hidden”

Wireless Auto Configuration Algorithm

- If still not associated and there is an ad-hoc network in preferred networks list, create the network and become first node
  - Uses self-assigned IP address (169.254.Y.Z)

Wireless Auto Configuration Algorithm

- Finally, if "Automatically connect to non-preferred networks" is enabled (**disabled by default**), connect to networks in order they were detected
- Otherwise, wait for user to select a network or preferred network to appear
  - Set card's desired SSID to random 32-char value, Sleep for minute, and then restart algorithm

Weaknesses in Wireless Auto Configuration

- *Information Disclosure*
  - Specific 802.11 Probe Requests reveal SSIDs of preferred networks
- *Spoofing*
  - Unencrypted networks are identified and authenticated only by SSID
- *Unintended Behavior*
  - An ad-hoc network in Preferred Networks List turns a wireless client into an Access Point

Positioning for Attack Against Wireless Clients

- Join ad-hoc network created by target
  - Sniff network to discover self-assigned IP (169.254.Y.Z)
- Create a stronger signal for currently associated network
  - While associated to a network, clients send Probe Requests for same network to look for stronger signal
- Create a (more) Preferred Network
  - Spoof disassociation frames to cause clients to restart scanning process
  - Sniff Probe Requests to discover Preferred Networks
  - Create a network with SSID from Probe Request

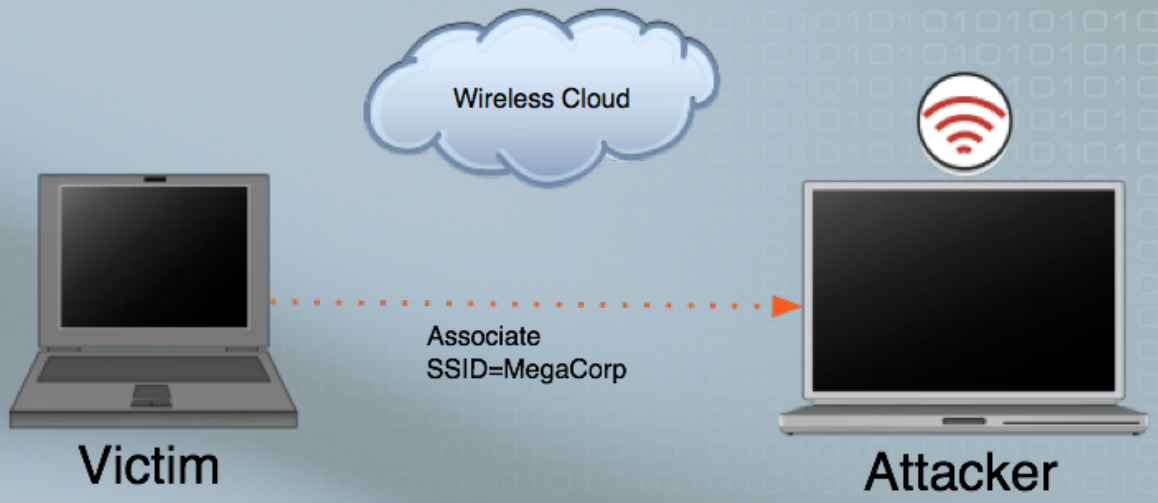
Attacking Wireless Auto Configuration

- Attacker spoofs disassociation frame to victim
- Client sends broadcast and specific Probe Requests again
  - Attacker discovers networks in Preferred Networks list (e.g. linksys, MegaCorp, t-mobile)

Attacking Wireless Auto Configuration

- Attacker creates a rogue access point with SSID *MegaCorp*

Attacking Wireless Auto Configuration



- Victim associates to attacker's fake network
  - Even if preferred network was WEP (XP SP 0)
- Attacker can supply DHCP, DNS, …, servers
- Attacker exerts a significant amount of control over victim

Improving the Attack

- Parallelize
  - Attack multiple clients at once
- Expand scope
  - Act as any networks that any client is looking for
- Simplify
  - Don’t require learning preferred networks before beginning attack
- Increase availability
  - Attack continuously

Attack Implementation

- Most wireless cards have firmware that enforce frame restrictions
  - Prism II HostAP mode doesn't pass Probe Requests to Operating System
- Atheros-based cards don't have firmware
  - Hardware Abstraction Layer (HAL) and all frame handling in driver software
- Attack implemented as modified Linux MADWiFi Driver
  - Respond to Probe Request frames for any SSID
  - Allow Assoc Request to any SSID

Performing The Attack

- Laptop runs software base station
  - Possibly with antenna, amplifiers
- AP responds to any Probe/Assoc Request
- Clients within range join what they think is one of their Preferred Networks
  - Client A thinks it is on “linksys”
  - Client B thinks it is on “t-mobile”
  - Client C thinks it is on “hhonors”
- Any client with at least one unencrypted preferred network will join if no legitimate preferred networks are present

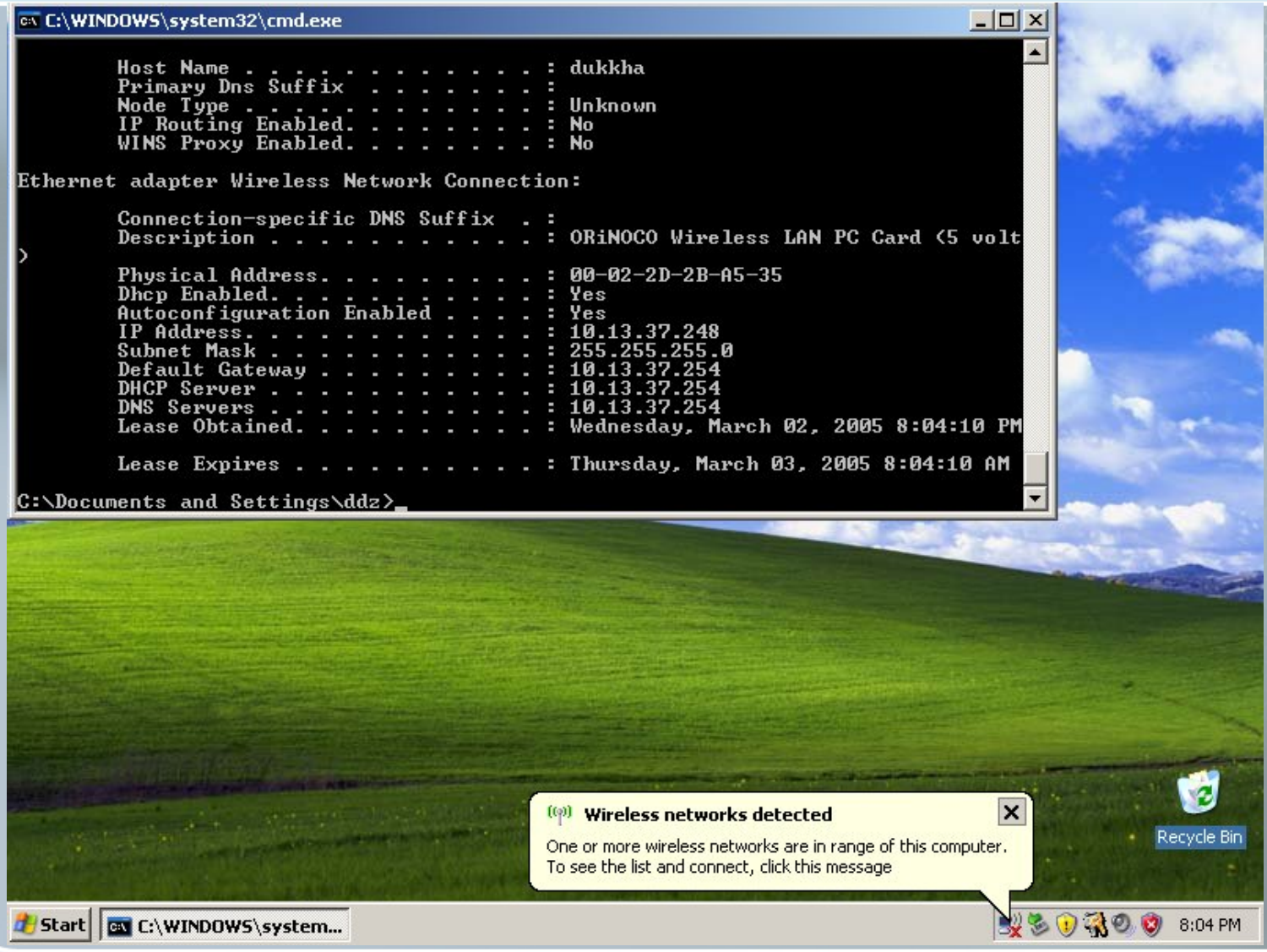
Wireless Auto Configuration Vulnerabilities

- Remember how SSID is set to random value?
- The card sends out Probe Requests for it
- We respond w/ Probe Response
- Card associates
- Host brings interface up, DHCPs an address, etc.
- Verified on Windows XP SP2 w/ PrismII and Orinoco (Hermes) cards
- Fixed in Longhorn

Packet trace of Windows XP associating using random SSID

1) 00:49:04.007115 BSSID:ff:ff:ff:ff:ff:ff DA:ff:ff:ff:ff:ff:ff SA: 00:e0:29:91:8e:fd Probe Request (^J^S^V^K^U^L^R^E^H^V^U...) [1.0* 2.0* 5.5* 11.0* Mbit]
2) 00:49:04.008125 BSSID:00:05:4e:43:81:e8 DA:00:e0:29:91:8e:fd SA:00:05:4e:43:81:e8 Probe Response (^J^S^V^K^U^L^R^E^H^V^U...) [1.0* 2.0* 5.5 11.0 Mbit] CH: 1
3) 00:49:04.336328 BSSID:00:05:4e:43:81:e8 DA:00:05:4e:43:81:e8 SA:00:e0:29:91:8e:fd Authentication (Open System)-1: Succesful
4) 00:49:04.337052 BSSID:00:05:4e:43:81:e8 DA:00:e0:29:91:8e:fd SA:00:05:4e:43:81:e8 Authentication (Open System)-2:
5) 00:49:04.338102 BSSID:00:05:4e:43:81:e8 DA:00:05:4e:43:81:e8 SA:00:e0:29:91:8e:fd Assoc Request (^J^S^V^K^U^L^R^E^H^V^U...) [1.0* 2.0* 5.5* 11.0* Mbit]
6) 00:49:04.338856 BSSID:00:05:4e:43:81:e8 DA:00:e0:29:91:8e:fd SA:00:05:4e:43:81:e8 Assoc Response AID(1) :: Succesful

“First of all, there is no ‘we’ …”



Vulnerable PNL Configurations

- If there are no networks in the Preferred Networks List, random SSID will be joined
- If all networks in PNL are encrypted, random SSID will have left-over WEP configuration (attacker will have to guess key)
  - We supply the challenge, victim replies with challenge XOR RC4 keystream
  - Our challenge is 0000000000000000…
  - We get first 144 bytes of keystream for a given IV
- If there are *any* unencrypted networks in PNL, host will associate to our modified Access Point.

Apple MacOS X

- MacOS X AirPort (but not AirPort Extreme) has similar issues
- MacOS X maintains list of trusted wireless networks
  - User can't edit it, it's an XML file base64-encoded in another XML file
- When user logs in or system wakes from sleep, a probe is sent for each network
  - Only sent once, list isn't continuously sent out
  - Attacker has less of a chance of observing it
- If none are found, card's SSID is set to a dynamic SSID
  - With 40-bit WEP enabled
  - … but to a static key
- After waking from sleep, SSID is set to "dummy SSID"
  - Will associate as plaintext or 40-bit WEP with above key
- MacOS X 10.4 ("Tiger") has GUI to edit list of trusted wireless networks

Defenses

- Keep wireless card turned off when not using a wireless network
- Only keep secure networks in Preferred Networks List
- Remove insecure network from PNL immediately after done using it
- Prevent mobile clients from connecting to sensitive networks

KARMA: A Wireless Client Assessment Tool

- Track clients by MAC address
  - Identify state: scanning/associated
  - Record preferred networks by capturing Probe Requests
  - Display signal strength of packets from client
- Allows targeting a specific client
  - Create a network they will automatically associate to
- Identify insecure wireless clients that will join rogue networks
- *"Kismet"* for wireless clients

KARMA Probe Monitor

KARMA

```
Hardware Address  Sig Probe Requests
00:11:f5:05:3f:bd 014 GoldenTree <broadcast>
00:02:2d:6e:7e:b2 012 GoldenTree
00:11:24:24:49:ff 007 GoldenTree
00:11:f5:0e:94:47 017 GoldenTree <broadcast>
00:90:96:a3:e0:f5 016 GoldenTree <broadcast> <random> Wayport_Access FortACE FortAce2
00:02:2d:6e:7e:a8 007 <random> NETGEAR <broadcast>
00:02:2d:36:9b:c9 018 <broadcast> GoldenTree <random> Agere Systems
00:90:96:f4:51:61 034 <broadcast> ^_^@ SMC Home Wireless default MIT loganwifi SCinet
00:12:f0:62:eb:ce 011 <broadcast> GoldenTree 101Co StataCenter GoldenTree jeremiah RC
00:05:4e:4e:10:8b 007 <broadcast> AirBears linksys Rwwatson Mercury FSL_Wireless etwi
00:11:24:97:ad:1c 011 <broadcast>nRouter AIRWAY ethostream GoldenTree antipodean
00:0e:35:82:83:f9 013 GoldenTree <broadcast>
00:12:f0:0a:c6:8f 003 <broadcast> HHS-AP4 FASTPASS GoldenTree ISAT-AP1
00:40:f4:ba:b8:a8 018 <broadcast>
```

Karma Attacks Radioed Machines Automatically

- Wireless and client-side attack and assessment toolkit
- Modules attack multiple layers as hostile server or Man-in-the-Middle
  - 802.11: Modified MADWiFi driver answers all Probe/Assoc Requests
  - DHCP: Rogue DHCP server points client at our DNS server
  - DNS: Rogue DNS Server responds to all queries with our IP address
  - POP3/FTP: Servers capture plaintext credentials
  - HTTP: Attack web server redirects any query to browser exploits or acts as transparent proxy

Conclusion

- Demonstrated weaknesses and vulnerabilities in Automatic Wireless Network Selection
  - Allows attacker to put victim on hostile subnet
- Firewalls commonly on by default, but clients still initiate a lot of traffic
  - Automatic updates
  - Browsing (NetBIOS, Rendezvous/Bonjour)
- Rise in client-side vulnerabilities
- Mobile clients are a risk to secure networks
- Assess risk of wireless clients with KARMA
  - http://www.theta44.org/karma/