Privacy-Preserving 802.11 Access-Point Discovery

Janne Lindqvist, TKK, Finland
Tuomas Aura, MSR, UK
George Danezis, MSR, UK
Teemu Koponen, HIIT, Finland
Annu Myllyniemi, TKK, Finland
Jussi Mäki, TKK, Finland
Michael Roe, MSR, UK

Outline

• IEEE 802.11 access-point discovery
• Privacy problem for the clients
• Possible solutions
• Privacy-preserving access-point discovery
Background 802.11 AP discovery

• AP initiated
  – Beacon

• Client initiated
  – Undirected active probe
  – Directed active probe

• Beacons and probes are used to discovery the presence of a network name, the SSID.
802.11 AP discovery: beacon

Beacon: SSID: MSFTWLAN

Beacon: SSID: MSFTGUEST
802.11 AP discovery

undirected active probe

Probe Request:
SSID is empty
802.11 AP discovery: 
*undirected active probe*

 Probe Response  
SSID: MSFTWLAN

 Probe Response  
SSID: MSFTGUEST

Diagram showing two routers broadcasting with overlapping signal strengths, indicating the discovery process of an undirected active probe.
802.11 AP discovery

directed active probe

Probe Request:
SSID: MSFTWLAN
802.11 AP discovery: directed active probe

Probe Response
SSID: MSFTWLAN
Discovery User Experience
“Hidden network”
discovery and association

1. **Beacon** (optional)
2. **Probe Request**
3. **Probe Response**
4. **Authentication Request**
5. **Authentication Response**
4-way handshake or EAP authentication

Hidden SSID (empty)
Probe for an SSID
SSID found
Messages containing the SSID

Broadcast
Broadcast or unicast
Unicast

AP
Client STA
“Hidden Network” User Experience
The Privacy Problem

• Clients keep a list of known networks, which they continuously probe.

• The SSIDs are plaintext identifiers
  – University, company, favorite Internet café
  – History of network usage
  – User fingerprinting/profiling [Pang et al., Mobicom’07]
Goal: Solving the Privacy Problem

• Protect the privacy of APs at least as well as in the current “hidden networks”
• Avoid the need for client to broadcast SSIDs when probing for “hidden networks”
  ➔ SSID not seen at all on air
• An observer can still see that some communication is taking place
Threat Model

• The adversary *can*
  – Move between network locations
  – Record and replay messages
  – Mount man-in-the-middle attacks at a *single* access point at a time

• The adversary *cannot*
  – Relay messages between two network locations (wormhole attacks)
Further constraints

• Deployability
  – No changes to the user experience
  – Cannot increase handoff latency
  – Minimal changes to 802.11 standard and implementations

• Must work together WPA-PSK or WPA2-PSK authentication
Possible solutions 1/3

- Remove the “hidden network” feature and require the APs to broadcast the SSID
  – This is not going to happen, because...
Possible solutions 2/3

• Use a random string as the SSID
  – Some implementations of WiFi Protected Setup actually do this
  – Not good for the user experience
    • SSID could be “¤#%!21%¤CXS)ASDF”
  – The user can still be profiled!
    • (possibly even better than before)
Possible solutions 3/3

- Probing not used as default, but needs to be manually enabled for each SSID
  - Windows Vista already does this
  - Users do not understand the tradeoffs
- Heuristics for reducing the number of probes
  - Heuristics often fail when the environment changes
  - Increases client implementation complexity
Our Approach

• Simple authentication protocol based on
  – cryptographic hash functions
  – symmetric key crypto
  – syntactically resembles ISO/IEC 9798-4

• Piggyback on the 802.11 undirected active probing
Privacy-preserving AP discovery

1. **Beacon** (optional)
2. **Probe Request**: $N_{\text{client}}$
3. **Probe Response**: $N_{\text{client}}, N_{\text{AP}}, E_{K_a}(R-\text{SSID}), \text{PRF}_{K_a}(N_{\text{client}}, N_{\text{AP}}, E_{K_a}(R-\text{SSID}))$
4. **Authentication Request**
5. **Authentication Response**
6. **Association Request**
7. **Association Response**
8. **4-way Handshake**

$K_a = \text{HMAC}_{PSK}(\text{"privacy key 1" } | N_{\text{client}} | N_{\text{AP}})$

$K_e = \text{HMAC}_{PSK}(\text{"privacy key 2" } | N_{\text{client}} | N_{\text{AP}})$

$R-\text{SSID} = $ pseudorandom value

$\text{PSK} = \text{PBKDF2}(\text{Password, SSID, SSID length, 4096, 256})$
User Experience

• The privacy-preserving discovery protocol does not use the SSID at any point.
• The SSID is configured as usual, so the client knows it
  – The user experience does not change
  – The name of the network is shown in the UI
Steps after Network Discovery

• WPA-PSK is privacy-preserving: continues with the 4-way handshake and encrypted connection

• Management frames need an SSID; we replace it with the R-SSID
  – new random R-SSID for each Probe Response
  – AP caches mapping between R-SSID and SSID for 60 seconds, longer if the client associates with it
Security Properties of the Protocol

• When a client probes for multiple APs, adversary cannot tell whether the APs belong to the same network or to different networks
  – (network = same SSID and PSK)
• When several clients probe for an AP, adversary cannot tell whether the clients have the same or different SSID/PSK
  ➔ stronger privacy protection than in current ”hidden networks”
• No changes to WPA-PSK security; we just reuse the PSK
Limitations

• WPA-PSK is privacy-preserving, but e.g. 802.1X authentication may leak the client identity
  – e.g. EAP-TLS send client certificate as plaintext
  – Would need to change the TLS handshake to have client identity protection
Performance Measurements

• AP implementation on Meraki Mini
  – Atheros AR2315 SoC @ 180 MHz
  – Runs Linux-based OpenWRT

• Client implementation on MadWifi drivers

• Measured also in the ORBIT testbed
  – PCs with 1 GHz VIA G3, 512 MB, Atheros AR5212 wireless interface
ORBIT Measurements

• We measured AP discovery latency
  – Compared to undirected active probing
  – Compared to hidden network discovery

• One AP

• 100 clients probed the AP every 125 ms each
ORBIT Measurements:
legacy broadcast vs. our protocol
Meraki Mini Microbenchmarks

- Measured the AP discovery latency with single client probing a single AP
  - Legacy WiFi: average 1.8 ms latency and median 1.5 ms
  - Our protocol: average 3.2 ms latency and median 3.1 ms
  - Replaced the cryptographic messages with constant data
    • average 2.8 ms latency and median 2.1 ms

- Raw processing times
  - Probe response created in 0.53 ms
  - Probe response verified in 0.34 ms
  ➔ Cost of cryptographic processing not an issue
Interesting observation on hidden network discovery

• Unexpected result from ORBIT measurements
• Current “hidden network” discovery implementations probe for one SSID on all radio channels, then try the next SSID
• Our protocol probes for all SSIDs with one challenge and all APs answer
  ➔ Our protocol is actually faster when the client probes for multiple SSIDs
Related work

- Impressive clean-slate design
  - Requires explicit pairing
  - Needs to consider clock skew
Further Information

• ACM WiSec’09 paper
  http://www.tml.tkk.fi/~jklindqv/wisec09web.pdf

• Further details in Microsoft Research Tech Report – MSR-TR-2009-07

• Source and patches coming to the web near you shortly.
Conclusions

• Small modifications to the standard WLAN
  – Co-exists with the current protocols and APs
  – Easy to deploy

• No changes to user experience
  – Configure like you would configure today

• *Enabler* for more complex privacy solutions such as MAC address randomization and other privacy mechanisms on upper layers.