WiFi Mobility with MPTCP

Vladimir Diaconescu
Costin Raiciu

University Politehnica of Bucharest

Context

- High AP density in urban areas
- Steady increase of WiFi-capable mobile devices
- WiFi is static by design
- Client is not taken care of when connectivity is lost

Current situation

- If the link fades, the client initiates a scan in order to find another AP
- Research: Speeding up the handover process
  - Does not solve the issue of losing TCP streams
  - Does not provide actual mobility

Our approach

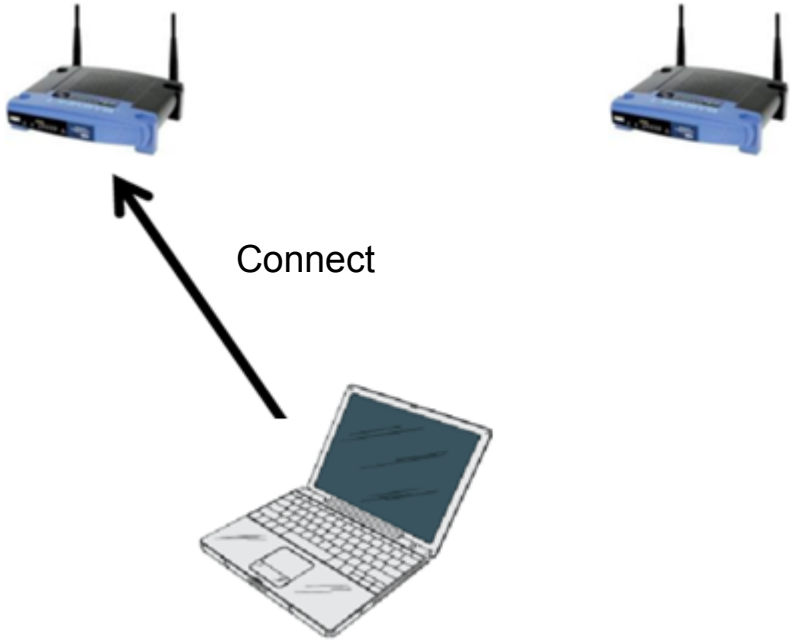
- Associate with multiple APs simultaneously
- Constantly switch between them (channels)
- Use virtual interfaces with different MAC addresses
- Use MPTCP over these, dynamically adding and removing streams

Channel switching

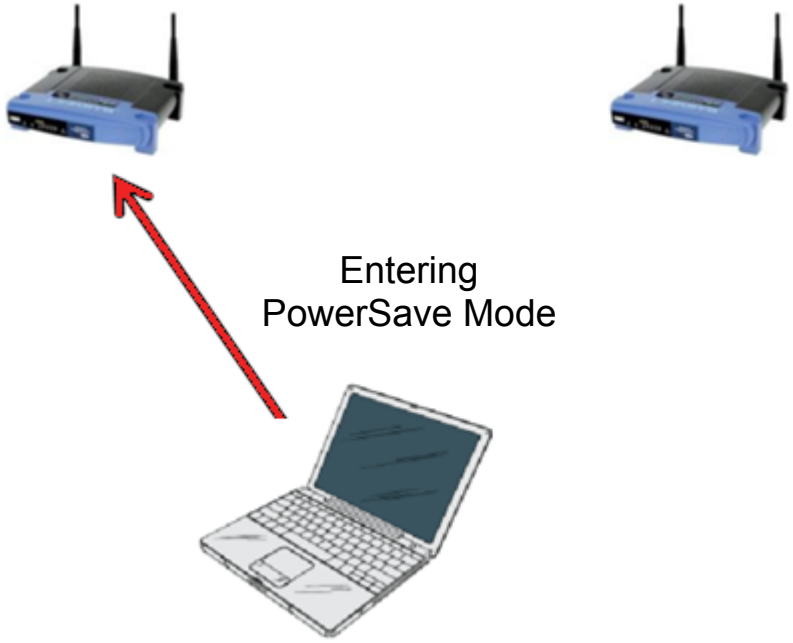
AP1

AP2

Channel switching



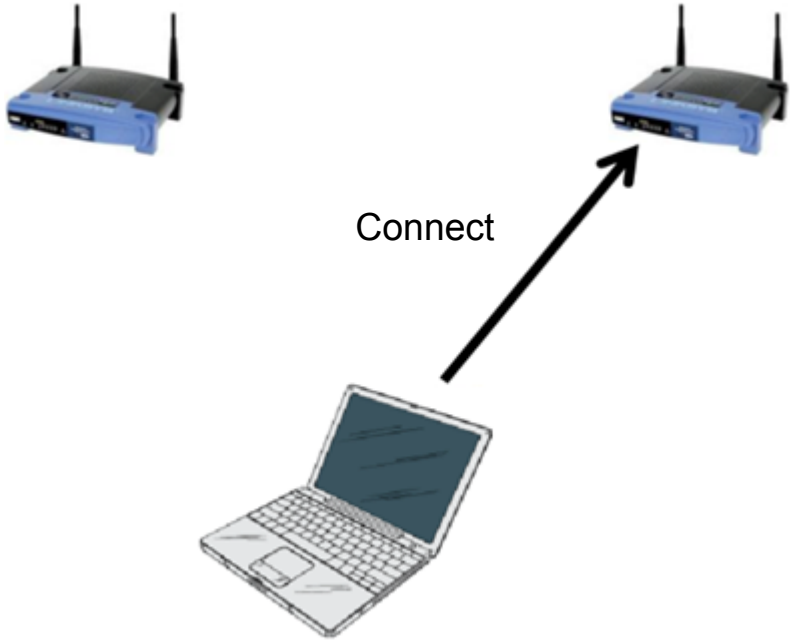
Channel switching



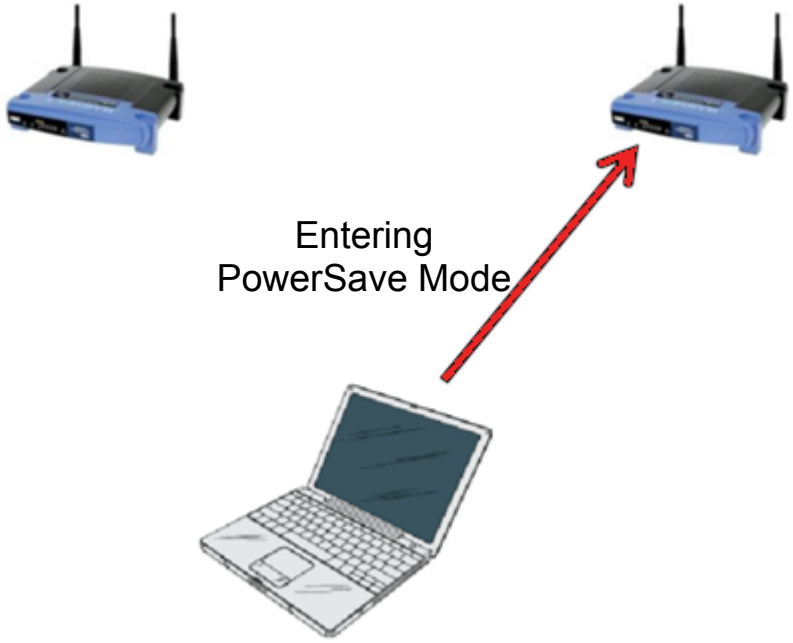
Channel switching

Reconfigure channel

Channel switching



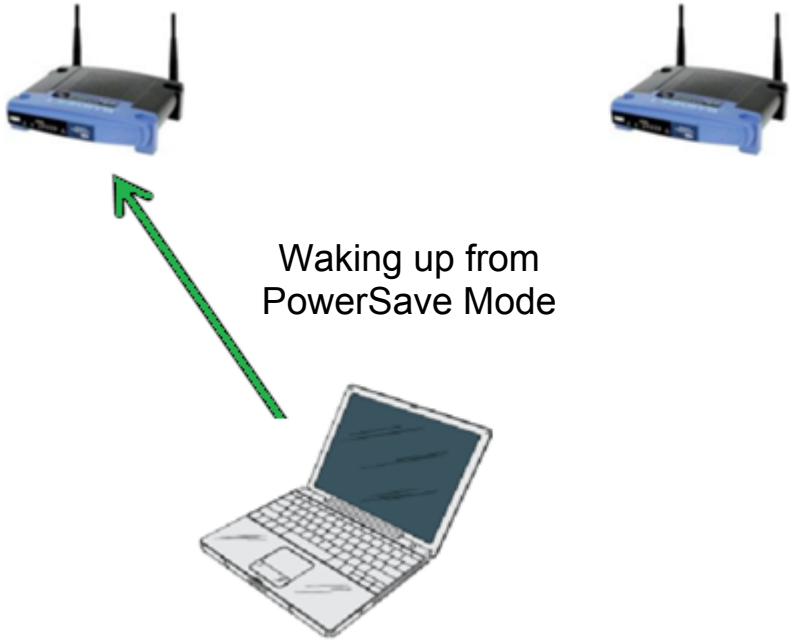
Channel switching



Channel switching

Reconfigure channel

Channel switching



Implementation

- Modified mac80211 kernel module
- Does not depend on the underlying driver or vendor
- Channel switching is triggered dynamically

Setting up

- MPTCP-0.89 (kernel 3.14)
- Compile modified mac80211
- Unload vendor-specific wireless driver(s)
- Unload mac80211
- Load custom mac80211
- Reload wireless driver(s)
- Configure routing

Things to hack on

Run mobility experiments for the whole week!

- Mapping the IETF access points- location, SSID, channel.
- Scripts to measure performance
- Kernel hacking to fix various issues

Questions