WiFi Networks Research Testbed for Commodity Routers

Manos, Igor

UPC, Guifi.net

August 3, 2015

What is WiBed?

WiBed is:

- A software platform aimed at deploying network experiments
- Also an OpenWRT-based platform to easily deploy and manage your mesh network
- Designed to run on commodity (cheap) IEEE802.11 routers
- Your best option for wireless networking experiments :)

What is WiBed?

but WiBed is also:

- An effort started by "hackers" in the WBMv6
- Complement Community-Lab.net testbed (Low Cost, Low layer Experiments)
- Fast-installed self-organized mesh network

Architecture Overview

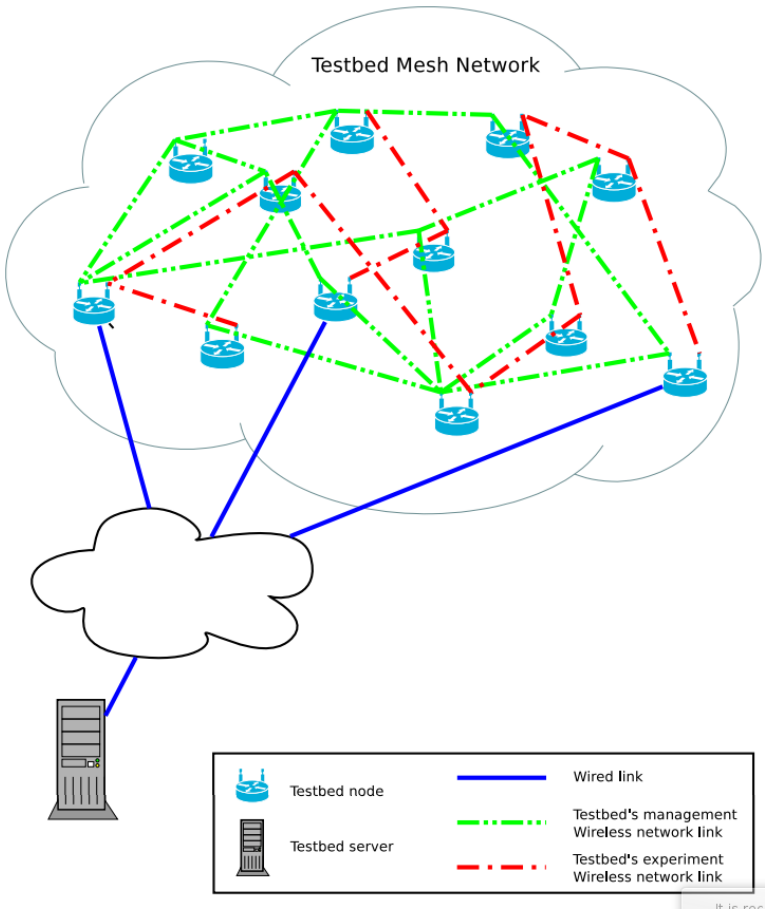

Figure : WiBed Architecture

Design Overview

- Nodes behave like FSM (idle-prepare-deploy-run-finish-idle)
- Communication with server through management mesh network
- REST-API pulling mechanism: every N seconds nodes pull state info and orders from server
- Node access mainly from the server web-UI
- Based on OpenWrt trunk
- Organized in packages, OpenWRT-compatible feed
- Management Network based on batman-adv

Hardware

- Node must be compatible with OpenWRT Linux (minimum 4MB flash)
- Node must have at least two radios (one for mgmt, one for experiments)
- Node must have at least one USB port (to store the overlay)

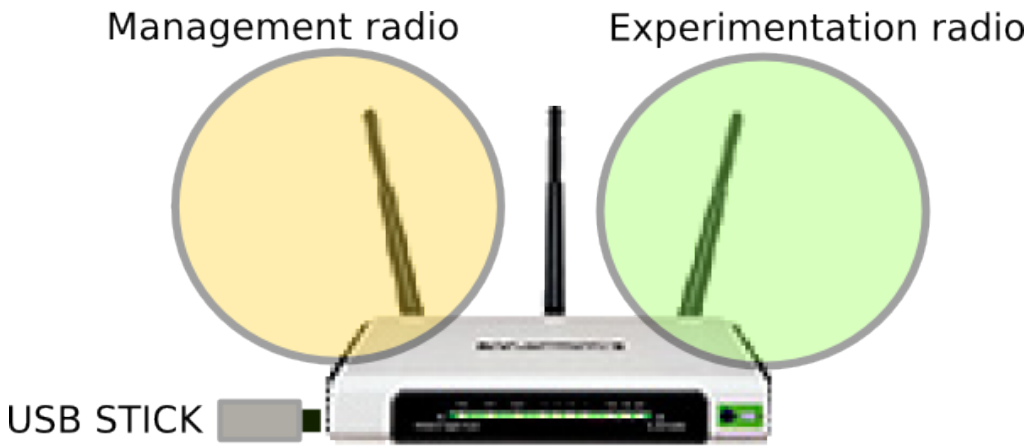

Figure : WiBed Node hardware

Diagram

- Nodes self-configure during the first boot
- IP address, hostname, ssid, etc. based on MAC address
- Experiments are overlays which are installed in the nodes
- Once an experiment finish, the overlay is removed and node goes back to initial state

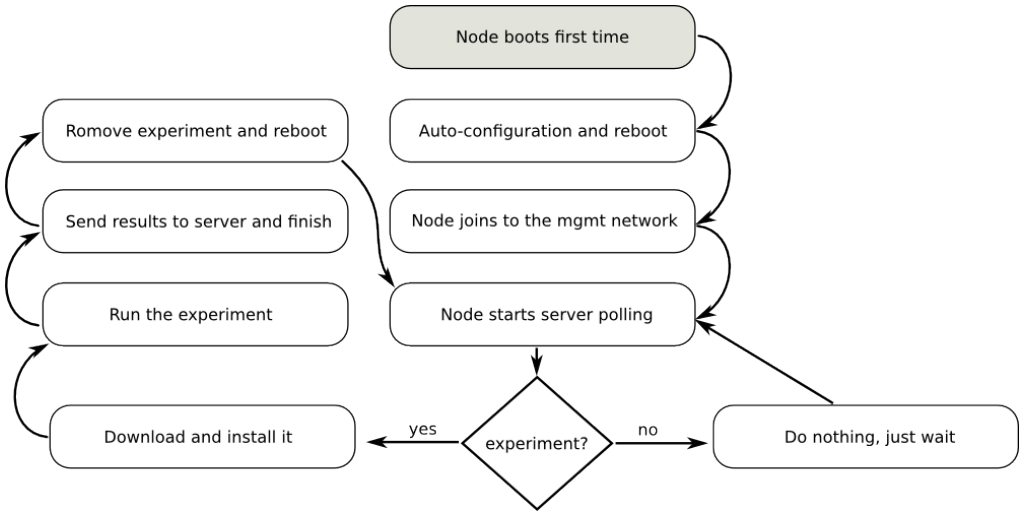

Figure : WiBed Node functional diagram

OverlayFS 1/2

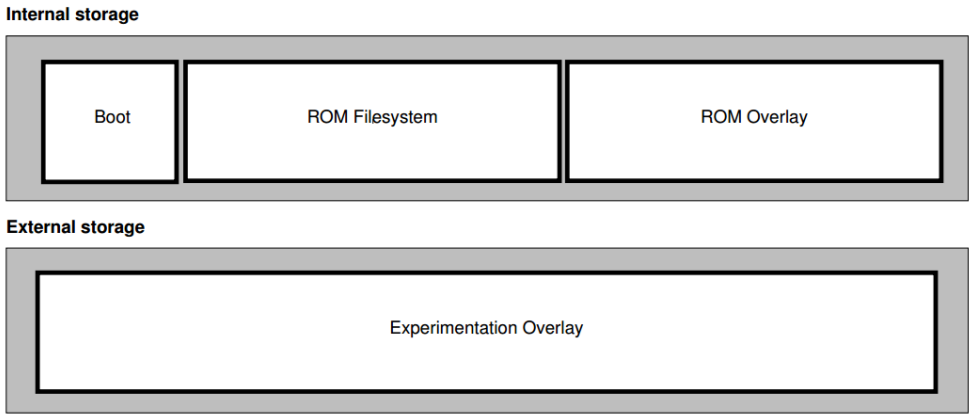

Figure : WiBed Node Filesystem

OverlayFS 2/2

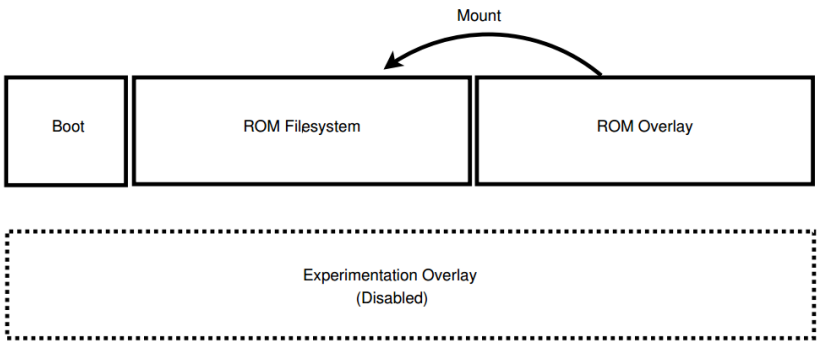

Figure : Node in IDLE state

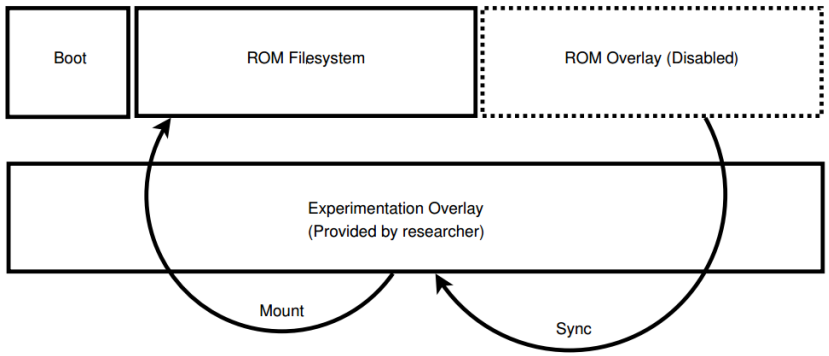

Figure : Node performing experiment

Config file

- WiBed uses UCI to manage the configuration
- It is flexible and allows many options
  For instance, management network device can be defined as "list ifaces radio2/radio1" meaning if radio2 exists, it will be used, otherwise radio1

```
config wibed general
    option node_id '00ff'
    option recovery_timer '00'
    option last_cmd_id '0'
    option command_id '00'
    option status '0'
    option api_url 'http://wibed.confine-project.eu/'

config wibed experiment
    option exp_id '0000'
    option ov_url 'http://wibed.confine-project.eu/send_ov'
    option save_url 'http://wibed.confine-project.eu/api/results'

config wibed upgrade
    option version '01'
    option model ''
    option auto '0'
    option upg_url 'http://wibed.confine-project.eu/upgrade'
    option upg_timer 'http://wibed.confine-project.eu/upgtimer'
```

```
config wibed management
    list ifaces 'eth0.1'
    list ifaces 'radio0/radio1'
    option channel5 '36'
    option channel2 '11'
    option bssid '02:C0:FF:EE:C0:DE'
    option ssid 'wibed'
    option ipv4_net '10.99.R1.R2'
    option ipv6_net 'fdba:99:R1R2::1/64'
    option ipv4_lan_net '192.168.R2.1'
    option country 'UZ'
    option txpower '20'
    option is_gw '0'
```

Figure : WiBed Node functional diagram

Wibed Server

- *Server*: `Tornado Web Server`
- *Our system*: `Flask` app + `SQLite`
- A REST API for interaction with nodes
- A web interface for interaction with users

Experiments

- Openwrt trunk: r46547
- Firmware: battlemesh/wibed
- Overlay: battlemesh/wibed-battlemesh-experiment
- Deployment Topology: Musti
- Experiments: Amadeus

Experiments

- Promising Kernel Patch?
- Protocol packages?

More info

- Wiki
- Paper
- Repo
- Us