🔍 WiFi Detection Fix & Test Guide

What Was Fixed

The WiFi detection on macOS was only checking `en0`, but on many Macs (especially Apple Silicon), the WiFi interface might be `en1` or `en2`.

Fixed:

- Now checks multiple interfaces: `en0`, `en1`, `en2`
- Better detection logic for active WiFi connections
- Added `hotspot test` command for diagnostics

How to Test

1. Rebuild the Project

```bash
cd ~/path/to/iptp
./build.sh
```

2. Run the Test Command

```bash
sudo ./dist/iptp-darwin-arm64

[IPTP-1] ~$ hotspot test
```

You should see:

```
=== WiFi Detection Test ===

🔍 Testing WiFi connection detection...
✓ WiFi is CONNECTED

🔍 Checking network interfaces...
Network Interfaces:
Hardware Port: Wi-Fi
Device: en0
Ethernet Address: xx:xx:xx:xx:xx:xx

🔍 Checking WiFi status for each interface:
  en0: Current Wi-Fi Network: YourNetworkName
  en1: Error or not available
  en2: Error or not available

🔍 Current IP addresses:
  inet 192.168.1.100 netmask 0xffffff00 broadcast 192.168.1.255

=== Test Complete ===
```

3. Test Auto Mode

Since you're **on WiFi**, `hotspot auto` should detect it:

```
bash

[IPTP-1] ~$ hotspot auto
🔌 Checking WiFi connection status...
✓ Already connected to WiFi
  Hotspot not needed

Your machine is currently on a WiFi network.
The DNS router is already running and can monitor DNS queries
from devices that use your machine's IP as their DNS server.
```

4. If You Want to Force Enable Anyway

```
bash

[IPTP-1] ~$ hotspot enable
🔌 Checking WiFi connection status...
⚠️  You are currently connected to a WiFi network
   Enabling hotspot may disconnect you from the network

Continue anyway? (y/N): y
# Then it will enable
```

What Should Happen (Based on Your Situation)

You're on WiFi, so:

Scenario A: Just DNS Monitoring (No Hotspot Needed)

```bash
sudo ./iptp
[IPTP-1] ~$ dns start
✓ DNS Router starting on 0.0.0.0:53

# Now configure devices to use your Mac's IP as DNS
# For example: 192.168.1.100
```

Scenario B: Want to Share Internet + Monitor

```bash
sudo ./iptp
[IPTP-1] ~$ hotspot enable -s "MyNetwork" -p "password123"
# This creates a hotspot even though you're on WiFi
# Other devices connect to your hotspot
# You can monitor their DNS queries
```

Scenario C: Smart Auto Mode (For When You're NOT on WiFi)

```bash
# When you're NOT connected to WiFi, run:
sudo ./iptp
[IPTP-1] ~$ hotspot auto
# This will automatically enable hotspot + DNS
```

Understanding the Workflow

When you're ON WiFi (like now):

1. Your Mac is connected to a WiFi network
2. You don't need a hotspot
3. But you can still run DNS router
4. Configure other devices to use your Mac's IP as DNS
5. Monitor their DNS queries

When you're NOT on WiFi:

1. Run `hotspot auto`
2. It enables your Mac as a WiFi hotspot

3. Automatically starts DNS router
4. Devices connect to your hotspot
5. Their DNS queries are automatically logged

Debug Output Explanation

The `hotspot test` command shows:

1. **WiFi Detection Result**: Connected or not
2. **Network Interfaces**: All available network ports
3. **Per-Interface WiFi Status**: Which interface has WiFi
4. **IP Addresses**: Your current network addresses

This helps diagnose why WiFi detection might fail.

Common Issues & Solutions

Issue: "WiFi is NOT connected" but you ARE connected

Solution: Check which interface your WiFi is on:

```
bash

networksetup -listallhardwareports
```

If it's on `en3` or higher, we need to add that to the check.

Issue: "Error checking WiFi status"

Solution: Check permissions and networksetup availability:

```
bash

which networksetup
networksetup -getairportnetwork en0
```

Issue: hotspot auto doesn't detect WiFi correctly

Solution: Run `hotspot test` and share the output. We'll add your WiFi interface to the detection list.

Quick Reference

Command	Purpose
`hotspot test`	Diagnose WiFi detection
`hotspot auto`	Smart mode (only if not on WiFi)
`hotspot enable`	Force enable (asks if on WiFi)
`hotspot status`	Show current status

Command	Purpose
`dns start`	Start DNS monitoring

Your Specific Case (On WiFi)

Since you're **already on WiFi**:

```bash
# Option 1: Just monitor DNS (simplest)
sudo ./iptp
[IPTP-1] ~$ dns start
# Configure devices manually to use your Mac's IP as DNS

# Option 2: Test the detection
sudo ./iptp
[IPTP-1] ~$ hotspot test
# See what it detects

# Option 3: Force create hotspot (if you want to share)
sudo ./iptp
[IPTP-1] ~$ hotspot enable
# It will ask for confirmation since you're on WiFi
```

The `hotspot auto` is designed for the scenario where you're **NOT** on WiFi - it automatically sets up everything. In your case, since you're already on WiFi, it correctly says "hotspot not needed".

Next Steps

1. Rebuild: `./build.sh`
2. Test: `sudo ./iptp` → `hotspot test`
3. Share the output if WiFi detection still doesn't work
4. We'll adjust the interface list accordingly

The fix should now correctly detect your WiFi connection on macOS!