Off Grid communications with Android

- Meshing the mobile world

m0nk and stoker have fun @ DefCon 20

Who are you guys?

- m0nk – Josh Thomas
  - jbthomas@mitre.org
  - m0nk.omg.pwnies@gmail.com
- Stoker – Jeff Robble
  - jrobble@mitre.org

- We work @ The MITRE Corporation (of CVE fame)

First off, let’s play a game

Where data goes to die

- Fukushima
- Katrina
- Haiti
- < Insert your “favorite” recent natural disaster here >
- Other?

Why do I care about Mesh networks?

- Physical infrastructure is prone to failure, networks shouldn't be
- Bypass the Cellular networks
- Bypass Wi-Fi networks
- Share information when infrastructure is broken or untrustworthy
- Extend and bounce other networks via bridging / tethering
- Headless

Ok, kind of cool. What about “Off Grid”?

- Single point of failure = single point of sniffing / filtering
- I don’t trust someone else being able to turn off my network, do you?
- When you want to share info, but don't want anyone watching ☺

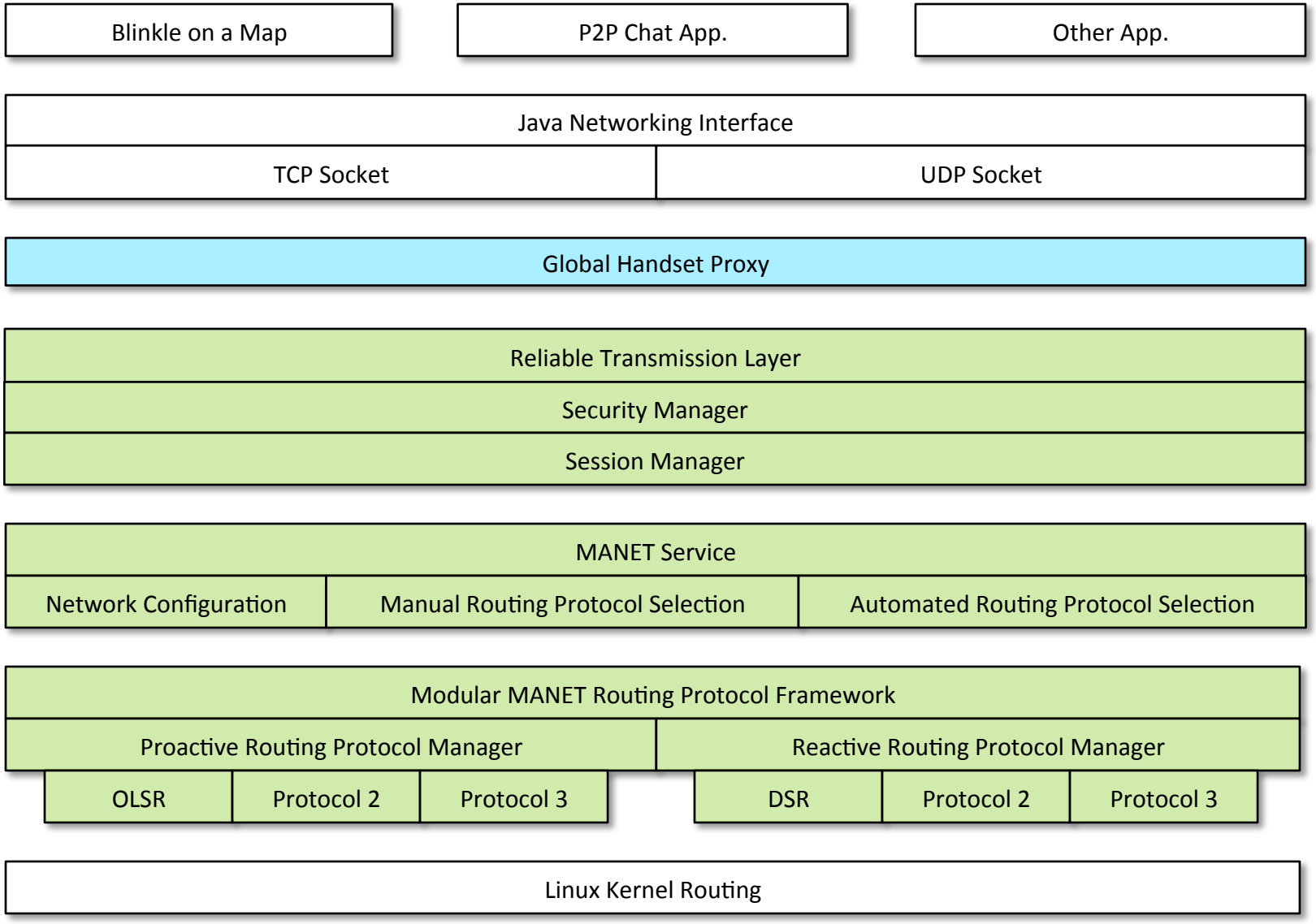
Your pocket contains more than a consumption device for Grumpy Fowl

- Wi-Fi chip with a fairly fat pipe
- Cell modem and baseband processor
- A ton of sensors
- (Somewhat) quality NAND and RAM
- A very under clocked and underutilized processor
- Power
- A boring screen that blinks!

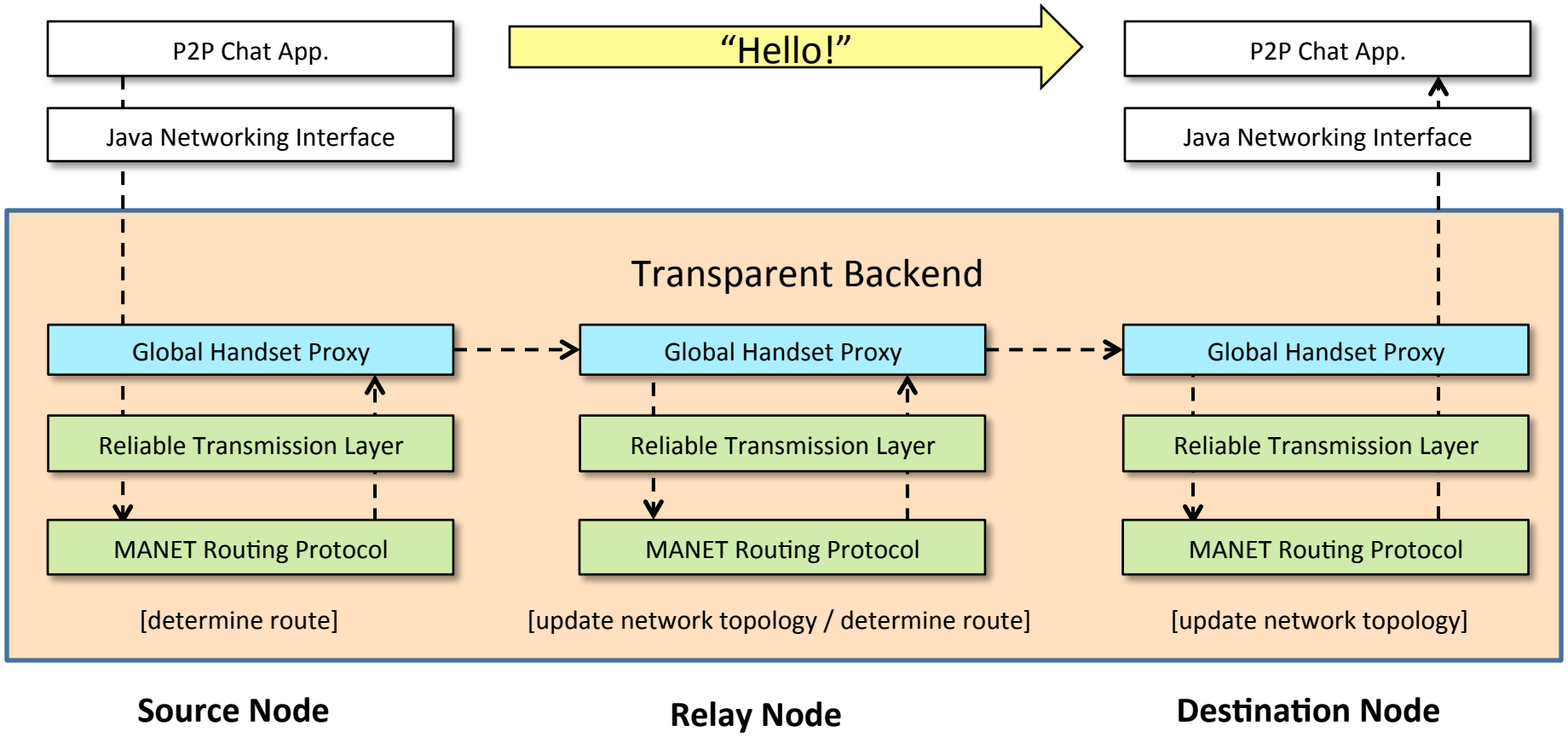
The SPAN framework

- We did the boring stuff so you don't have to!
- General Overview of the framework, what / why / how
  - Harnessing SPAN for your own project?
  - Repurpose root to muck with your WiFi chipset

SPAN + Android Technical Architecture



Data Flow


P2P Chat App.
Java Networking Interface
"Hello!"
P2P Chat App.
Java Networking Interface
Transparent Backend
Global Handset Proxy
Reliable Transmission Layer
MANET Routing Protocol
Global Handset Proxy
Reliable Transmission Layer
MANET Routing Protocol
Global Handset Proxy
Reliable Transmission Layer
MANET Routing Protocol
[determine route]
[update network topology / determine route]
[update network topology]
Source Node
Relay Node
Destination Node

A Deeper dive into the Android Network stack implementation

- Thank you Harald Mueller
- I don’t want to be in managed mode
- Wireless Extensions API and support
- Pre and post ICS

Why we love Broadcom

- Flipping chipsets into Ad-Hoc Mode

Device	**Wireless Chip**
Samsung Nexus S 4G	Broadcom BCM4329
Samsung Galaxy Tab 10.1	Broadcom BCM4330
Samsung Galaxy S II Epic Touch 4G	Broadcom BCM4330
Samsung Galaxy Nexus	Broadcom BCM4329
ASUS Eee Pad Transformer Prime	AzureWave AW-NH615 (rebranded Broadcom BCM4329)
Motorola Razr Maxx	Texas Instruments WL1285C
iPhone 4S	Broadcom BCM4330
Nokia Lumia 900	Broadcom BCM4329

Kernel v. Metal

Wireless Extensions Support	No Wireless Extensions Support
Samsung Nexus S 4G	Samsung Galaxy Nexus
Samsung Galaxy Tab 10.1	ASUS Eee Pad Transformer Prime
Samsung Galaxy S II Epic Touch 4G	Motorola Razr Maxx

- Dear Vendors: Please either stop mucking with your kernel source or provide it to the community.

Plug and Play / Dynamic routing algorithms and you!

- Adjusting packet routing at runtime, a 5 minute primer on untrustworthy routing tables
- The tradeoffs of Bandwidth vs. Network Scale and Multi-Hop headaches
- File share, Chat, Disconnected Twitter and VOIP over a Mesh. Oh, the fun we can have.

This slide should not be needed

- What do I use a network for?
  - Chat
  - Data and file sharing
  - VoIP
  - Situational Awareness and Crisis management
  - Disconnected Twitter

OLSRd

- Object Link State Routing daemon
- Great project and Open Source
- Proactive protocol
  - Manage the mesh with simple hello packets
  - More overhead than we like
- Lots of knobs to turn here

Simple with Dijkstra

- Still proactive
  - But with almost unlimited knobs for tuning the mesh
- Less chatter over the Air

Reactive Protocols

- Stale routing table = What routing table?
- No we can play with motion and location in a useful way
- Don’t forget that if you pack node location into the headers it can been seen by others
- Downsides come with throughput issues

An aside on Delay tolerance

- Disconnected nodes act as disjoint message queues
- The protocol thinks of the device as a carrier pigeon (RFC 2549)
- Fall back to message passing

Scale, Delay and Hopping

- Though we see great improvements, simple proactive routing uses a ton of bandwidth to stabilize the network
  - Still, we can predict bandwidth and throughput metrics
  - VoiP good until we scale quite large
- Reactive routing has less chatter with the same bandwidth but is laggy
- Mix them FTW.

More Tunnels and some preliminary Security

- Jumping over the cell network or Wi-Fi (Mimicking VPN with standard Tunnels)
- Tunneling the mesh through the Internets!
  - VPN clusters and remote enclaves
- Securing the mesh from unwanted guests
- Jumping through unsecured mobile nodes

Jumping over the cell network or Wi-Fi

- Your device has 2 network ports (Wi-Fi & Cell):
  - We can connect them
  - We can bridge them
- Tablet with no cell chip?
  - Plug in an Alpha
- Virtual mesh networks connected using simple VPN tunnels

A Security Paradigm?

- Use Bluetooth or NFC to Bump transfer configuration info and keys
- Secure each link / node with its own keys
- Encrypt network data such that bounce or hop nodes cannot decrypt

- ICS & Wi-Fi Direct: Meshing internals

- Why do I have 10 MAC addresses and can I change them?
- Initial ICS drop is a very lame partial implementation of the spec
- Possible upgrade in JB?

Sexier Android Deployment

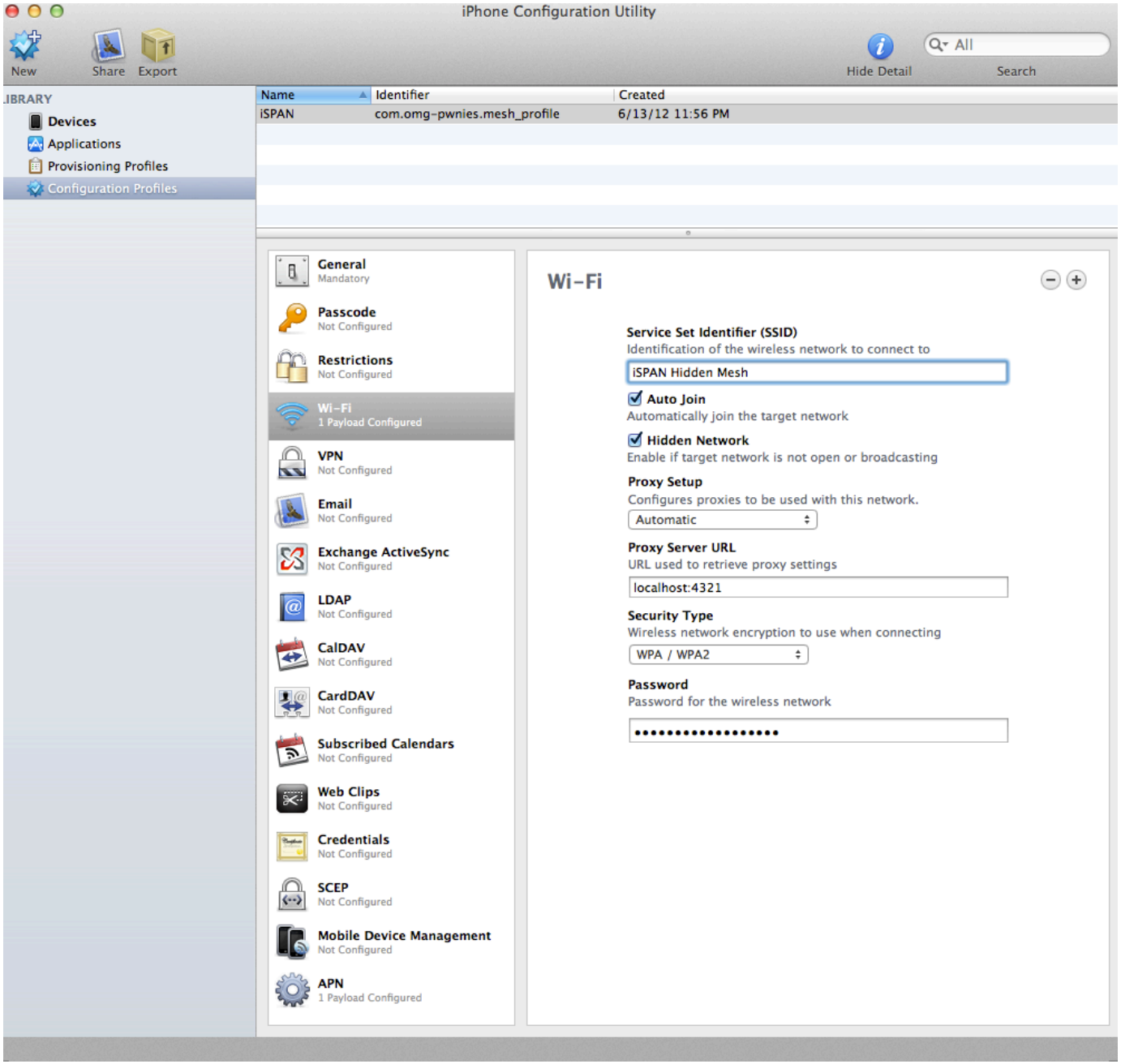
- We don’t need root forever, just install
- Grab Zerg, wrap in APK and pop the phone on install
- Root goes away - mesh stays
- Over the Air install?

What about my…?

- A:
  - iPhone: In Theory
  - Black Berry: Maybe?
  - Windows Phone: Yes (why do you own one?)
  - Arduino / GumStix: Yes
  - Netbook / Linux / Mac / Windows Box: Yes
  - Toaster: Yes but Why?
- Framework is a mix of Java and C
  - If your box can run those…

iOS?

- Apple gave us a built in Wi-Fi proxy configurable with the iPhone Configuration Utility
- Ooohhh, is that an APN setting as well?
- Cool, now all we need is a simple server to proxy and route our data


iPhone Configuration Utility
New
Share
Export
Hide Detail
All
Search
LIBRARY
Devices
Applications
Provisioning Profiles
Configuration Profiles
Name
Identifier
Created
iSPAN
com.omg-pwnies.mesh_profile
6/13/12 11:56 PM
General
Mandatory
Passcode
Not Configured
Restrictions
Not Configured
Wi-Fi
1 Payload Configured
VPN
Not Configured
Email
Not Configured
Exchange ActiveSync
Not Configured
LDAP
Not Configured
CalDAV
Not Configured
CardDAV
Not Configured
Subscribed Calendars
Not Configured
Web Clips
Not Configured
Credentials
Not Configured
SCEP
Not Configured
Mobile Device Management
Not Configured
APN
1 Payload Configured
Wi-Fi
Service Set Identifier (SSID)
Identification of the wireless network to connect to
iSPAN Hidden Mesh
Auto Join
Automatically join the target network
Hidden Network
Enable if target network is not open or broadcasting
Proxy Setup
Configures proxies to be used with this network.
Automatic
Proxy Server URL
URL used to retrieve proxy settings
localhost:4321
Security Type
Wireless network encryption to use when connecting
WPA / WPA2
Password
Password for the wireless network
••••••••••••••••••

What else can we use the Mesh for?

- Mobile data redundancy using the Torrent protocol to raid data across all devices?
- Distribute threads and tasks across a cloud of unused processors?
- Spoofing?

Dumb enough to attempt a demo!

- Oh wait, we already did?