Kismet

Dragorn
dragorn@Kismetwireless.net

July 10, 2004

Kismet

- 802.11 sniffer
- Can sniff 11a, 11b, and 11g with the right hardware
- Completely passive
- Signature and Trend Layer2 IDS
- Runs on Linux, some BSD, OSX
- Plays nice with the other kids on the block (snort, driftnet, etc)
- Can log GPS coordinates for mapping of network positions
- Can run on extremely lightweight devices (APs, handhelds, etc)
- Client/server so multiple displays can connect to one server, custom clients can perform additional logging, GUIs, etc

What's in a name?

- Go to dictionary.com
- Click synonyms until bored
- ??
- Profit

Scratch an itch

- Most common reason any free software gets developed
- No sniffers specifically for wireless to handle detecting networks (at the time)
- Started as modifications to the original 0.1 airsnort for prism2 only
- Added support for other chipsets and it grew into its own project with a life of its own

Supported platforms

- Linux - Core development is on Linux, most supported platform.
- OSX - Original airport cards work, Airport Extreme will not. For a pretty GUI, check out Kismac
- *BSD - Prism2 cards in OpenBSD, Atheros and Prism2 on FreeBSD5, unknown on NetBSD. Freebsd Radiotap system will bring more support
- Windows - Currently no free drivers for doing monitor mode. Kismet can read from WSP100 embedded sniffers, or Kismet drones on a supported platform.

Sniffer vs. Stumbler

- Sniffers like Ethereal, Tcpdump, or Kismet capture raw data frames. Kismet always operates in monitor mode, other sniffers can. Sniffers can see data packets.
- Stumblers query the card firmware to see what networks are detectable in the area. They usually see fewer networks than sniffers, and can't capture data packets, but they don't require special drivers, either.

RFMon vs Promisc

- Monitor mode (rfmon) puts the card into a state that is not connected to any network, and will report all packets including management frames to the OS.
- Promisc mode has less meaning in wireless - with most drivers, will report all data frames on the associated network, but not from other networks or management frames
- Monitor mode requires support from the chipset and the driver. Most chipsets can do it, but not all drivers support it on all platforms.

Passive network detection

- Monitor mode gets us all management frames
- Management frames define the network
- Able to directly detect the presence of APs and infer the presence of hidden networks from other traffic
- Able to decloak hidden SSIDs by watching client connections
- Passive network detection just that - passive. No packets are sent by the sniffer
- Passive sniffers can also detect active sniffers like Netstumbler

Demo

Kismet demo goes here

Detecting hidden networks: Security that isn't.

- Vendors try to add security by modifying the protocol, but it really has zero gain
- Cloaked SSID: APs don't put the SSID in the beacon frame. This is supposed to prevent people who don't know the network name from connecting, but the SSID was never designed to be a security feature, and is sent cleartext by the AP when a client joins.
- Nonbeaconing: Some APs attempt to turn off or slow down the beacon rate so that they're invisible, but as soon as a user exchanges data on the network, it can be seen.
- Hiding networks from passive sniffers is impossible, as long as the sniffer is capable of understanding the protocol and listening on the frequency.

Wireless IDS

- Easiest is fingerprint matching - some tools always send a certain frame which is indicative of an alert condition
- Netstumbler, Lucent Site Survey, Wellenreiter, and some 802.11 layer attacks are all fingerprint based.
- Trend based alerts detect events which are normal in small amounts or in some sequences, but constititute attacks in other situations.
- Flooding, AP spoofing, and generic active sniffer detection are all trend-based alerts

Distributed IDS

- Kismet drones are a super-stripped down version of the Kismet core which report packets over a wired network
- Even more lightweight than the server
- Runs well on 486s, APs, etc
- Distributed net of drones linked to one server running WEP decryption and IDS
- An entire building can report to one Kismet engine for logging and IDS

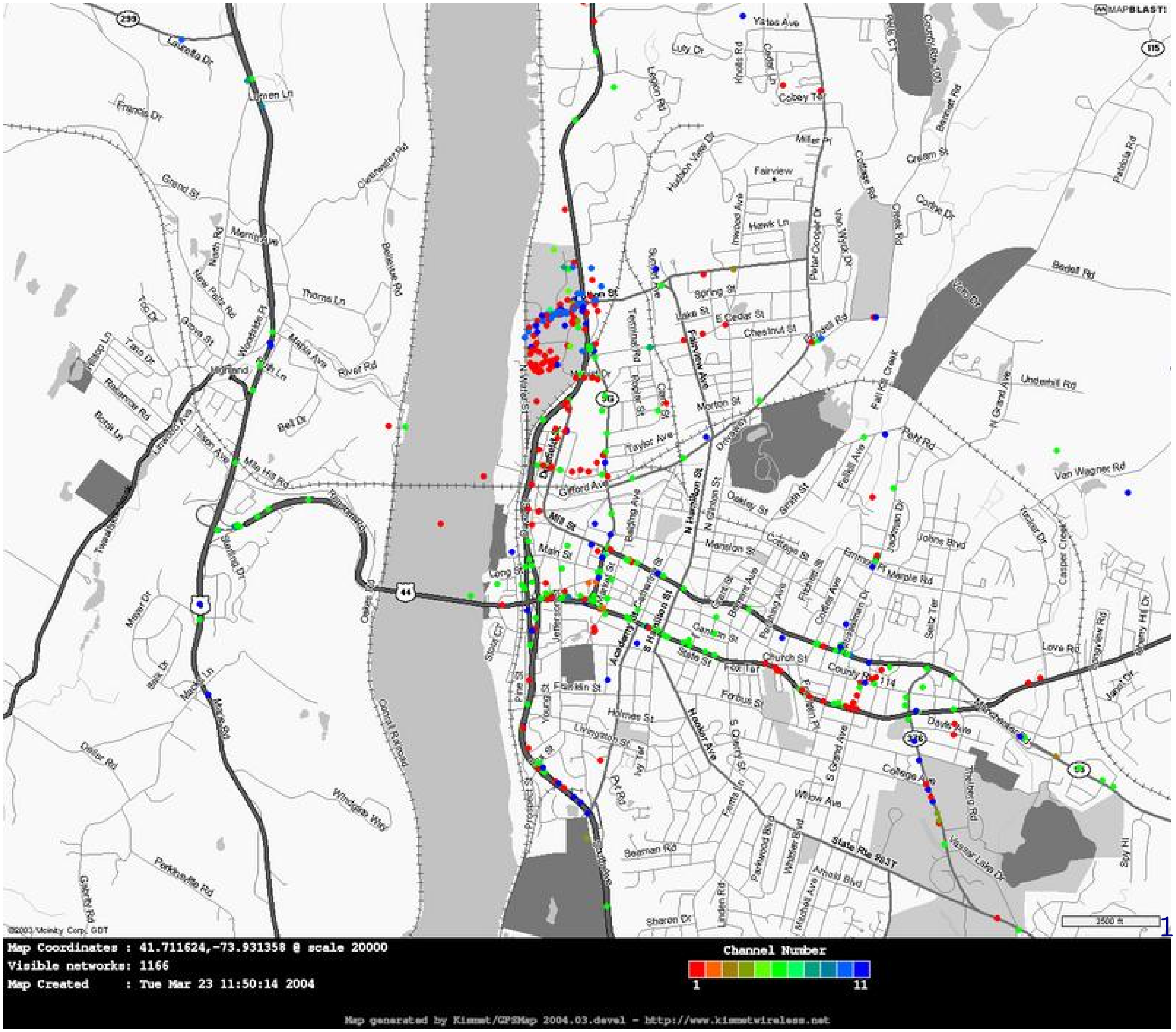
Cooperation with other tools

- Unix philosophy - Smaller tools that work with other tools
- Kismet dumpfiles are standard tcpdump format - any tool that can read pcap files can read a Kismet dump, ie tcpdump, ethereal
- Live packet streaming via FIFO pipe for other tools, including dewepping of packets with known keys.
- FIFO pipe allows tools like Snort to attach to the stream of packets processed by Kismet and perform layer3+ IDS functions.
- Entire building-wide drone network can be routed to snort for TCP IDS
- XML logfiles for external parsers to reprocess network data for web sites, databases, audit logs, etc

GPSMap

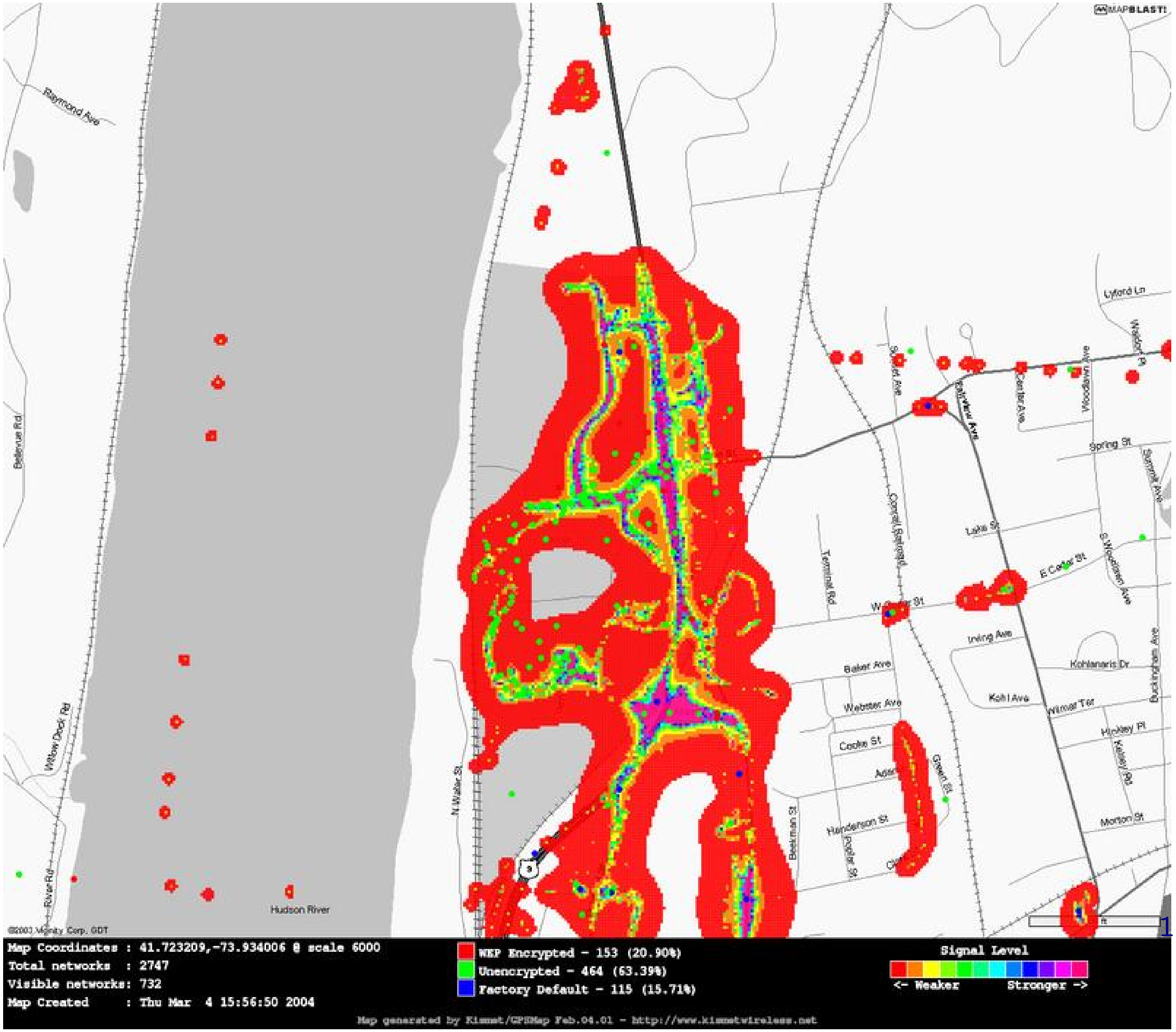
- Processes gpsxml and netxml files generated by Kismet to plot on graphical maps
- Pulls from various public map sources
- Does extensive data sifting to clean up bad sample data caused by GPS glitches
- Sample grouping and averaging to find the “most likely” set of sample points for the center of the network
- Several drawing modes for different visualizations of the data: Estimated network range, interpolated signal level graphing, travel path, estimated network center point, convex hull of all sample points, etc. Networks can be colored by channel or WEP type

AP map



```
Map Coordinates : 41.711624,-73.931358 @ scale 20000
Visible networks: 1166
Map Created     : Tue Mar 23 11:50:14 2004
```

Signal level map



```
Map Coordinates : 41.723209,-73.934006 @ scale 6000
Total networks  : 2747
Visible networks: 732
Map Created     : Thu Mar  4 15:56:50 2004
```

🟥 WEP Encrypted - 153 (20.90%)
🟩 Unencrypted - 464 (63.39%)
🟦 Factory Default - 115 (15.71%)

Signal Level
<- Weaker   Stronger ->

Map generated by Kismet/GPSMap Feb.04.01 - http://www.kismetwireless.net

Cards that work well

- Prism2/2.5/3 - Excellent 11b card that is very well understood with good drivers. Best chipset for wireless hacking
- Orinoco - Old Orinocos work very well. New Orinocos have changed to HermesII which cannot yet do rfmon
- Atheros - 11a/11b/11g chipset with good general and monitor support
- PrismGT - Monitor capable drivers in Linux

Cards that work

- Airport - OSX airport rfmon drivers work but can be touchy
- Centrino - Drivers have rfmon in Linux, but currently report invalid packets with no method to validate them
- Cisco - Hardware is good, drivers are unreliable
- Acx100 - Drivers with rfmon for some platforms
- Admtek - Binary drivers for Linux with a monitor mode hack, GPL drivers under development

Cards that don't work

- Broadcom - many cards use a broadcom chipset (Airport extreme, new Linksys, more. Broadcom will not release specs to write open drivers.
- Airport Extreme - Rebadged broadcom
- Atmel - Atmel cards have no monitor ability in the chipset
- Realtek - Primarily software driven, but no monitor support
- HermesII - No monitor support yet

Other tools

- Netstumbler - http://www.netstumbler.com - Windows stumbler
- Kismac - http://www.binaervarianz.de/projekte/programmieren/kis - OSX native passive sniffer similar to Kismet
- Ethereal - http://www.ethereal.com - Graphical general sniffer for many platforms
- Tcpdump - http://www.tcpdump.org - Text-based general sniffer for many platforms