802.11 Wireless Networking Deployment Survey for Dublin, Ireland

Niall Murphy (a),
David Malone (b)
and Ken Duffy (b)

September 2002

(a) Enigma Consulting Limited, 45 Dawson Street, Dublin 2, Ireland

(b) Communications Network Research Institute, DIT, Dublin 6, Ireland

Abstract

This document describes the results obtained from, and the experiences of, a survey of wireless networking deployment in Dublin, Ireland conducted between May and July 2002. Recent advances in wireless technology, and its standardization, enable low-cost alternatives to traditional wired services for both commercial and non-commercial users; from omni-direction local area "hot spots" to long-distance, line-of-sight, point-to-point links. However, the adoption of wireless networks leads to security issues not encountered with traditional wired networks.

We take Dublin as a case-study of wireless networking deployment, and four distinct aspects are focused on: the geographical distribution and volume of wireless equipment, the density of networks not running WEP-based security, the breakdown of equipment currently in use, and the applicability of a range of off-the-shelf equipment for sensing wireless networks.

We found that a large number of both local area and point-to-point wireless solutions are being employed throughout Dublin, a large fraction of network operators are not addressing wireless security at the link layer, a wide range of manufacturers' equipment is being used, and we identify a practical setup for sensing wireless networks.

Table of Contents

---

1. Introduction

1. Motivation for survey

The primary objective was to investigate the level of deployment of wireless networks in Dublin, a typical European city. We were also interested in the amount of attention being paid to the security implications of wireless networks, this being measured by the number of access points having some form of security enabled, whether WEP-based or not. We hope this survey will illustrate issues that occur in the real world, when the technology is used by real users, issues that may not have been anticipated during the design phase of 802.11.

All techniques used during the survey are publicly available. It would be straightforward to reproduce our results and, if one was so inclined, misuse them. Non-transport-layer-encrypted wireless networks, even those using WEP [3], are more susceptible to passive eavesdropping than their wired equivalents. We strongly advise those using wireless equipment to pay particular attention to current best practice in security.

2. Introduction to 802.11 wireless networks

For a thorough introduction, see Mathew S. Gast's book [4].

The growing take-up of wireless networking has, in part, been made possible by market crystallization around 802.11 as the definitive standard for sharing resources over the air.

The name 802.11 derives from the 802 family; a name chosen by the IEEE for networking standards, both physical and logical. Within 802.11 there are numerous sub-standards. Of them, 802.11b is, by far, the most commonly deployed, enabling a maximum transmission speed of 11Mb/s. All these standards control how network-cards should transmit data. They also have mechanisms for dealing with interference and for controlling interaction between access-points and clients.

There are features pertinent to wireless networking, which have been designed into the protocol, that are not relevant in wired networking. To control access to access-points, a mechanism called *association* exists. This is a process whereby a client may "register" with an access-point to gain access to its wireless network. There is the potential to use a built-in encryption standard, relying on a shared key, called Wired Equivalent Privacy (WEP).

Some nomenclature

A group of wireless devices (say, laptops with 802.11b cards) communicating directly to each other without central control, is called an *ad-hoc network*.

An *access point* is a central device which controls access to the wireless medium. It is, typically, also connected to a wired network.

An access point, which might have several clients (say, laptops with 802.11b cards), is called an *infrastructure access-point* or an *infrastructure Basic Service Set (BSS)*.

A group of access-points within a single organization, configured to allow clients move between access-points while maintaining connectivity, is called an *Extended Service Set (ESS)*. This whole system (access points, plus back end networking glue to connect them together) is called a *distribution system*.

Wireless-Fidelity (WiFi) is an industry-sponsored synonym for 802.11 compatible equipment.

Scanning for networks

Legitimate clients recognize that a wireless network is present by using a procedure called scanning, which is either active or passive. Scanning involves interacting with management frames called beacon and probe frames. Beacon frames are sent by access points periodically. They contain network- and radio-relevant information that allow the client to connect.

Active scanning sends out a probe request frame on each available channel. Probe response frames are sent back as acknowledgments by access points on those same channels. As with beacon frames, these response frames contain management information necessary for successful connection to the wireless network. It is possible to send out a probe request to the "broadcast" SSID which, in theory, provokes a response from every nearby access point. Certain access points can be configured not to respond to probe request frames through the use of MAC address filtering or other "cloaking" features.

During passive scanning, a card will hop from channel to channel listening for

beacon and probe response frames. If a network does not transmit such frames during the time that the card is listening, it will not be observed. However, networks that don't respond to the broadcast probe request frames can be found using this method, as long as there is ongoing traffic using the medium.

The active scanning procedure is roughly equivalent to the method NetStumbler uses to detect networks on Windows. Passive scanning is used by Kismet. We used both pieces of software and hence both methods in the course of this study.

3. Security problems with wireless networks

Previous studies [7] have demonstrated the security weaknesses in various aspects of 802.11 networking. Most have concentrated on WEP, a poorly designed protocol intended to provide "more or less" the same level of privacy as wired networks. Aside from WEP, there are other security considerations that wireless networking introduces.

- **Ease of access to medium**

  The distance from an access point at which a wireless network can be sensed depends on the particular protocol employed and on physical barriers. For instance, the typical effective radius of 802.11b in an unobstructed environment is 100 metres. Wireless signals can radiate beyond the bounds of intended users. This is a serious issue as, for example, in a business park hosting two commercial rivals, the wireless network of one may extend into the premises of the other.

- **Ease of interception**

  The observation of traffic not intended for you, known as interception, is clearly a problem in a broadcast medium. This problem exists in traditional wired networks, but is more obvious in wireless networks. For example, in wired networks, switches ensure that packets destined for one computer are not seen by others. These switches can be fooled: some switches do not adequately manage their list of MAC address to port mappings; when inundated with new MAC addresses, some fail in a mode that broadcasts all packets out of every port. Suites of tools, such as dsniff [2] have been written to demonstrate these vulnerabilities.

- **Ease of injection**

  Injection, in this context, means being able to create invalid packets that might be accepted by a station as being valid. This can be done by overriding or replacing existing packets sent during a legitimate session. Cryptographically protecting packets is the best defense against this attack.

- **False access points**

  False access points are readily introduced into wireless networks. They claim to be part of an organisation's network and intercept traffic. This is similar to the false base station attack found in GSM [14].

- **WEP**

Wired Equivalent Privacy has serious weaknesses. If you collect enough packets, it is possible to break WEP encoding and obtain a private network's shared, secret key. Since frames use LLC encapsulation, the first byte is always *oxaa*. This enables a "known plain-text" attack.

- **802.1x, EAP and LEAP**

  802.1x, an extension to 802.11 that aims to have greater security than 802.11b, uses the Extensible Authentication Protocol (EAP). EAP is a protocol designed for pluggable authentication, specified in RFC 2284 [11]. It aims to make having a single network sign-on more simple and generic. The sign-on allows access to a given network port and was designed in (as the name suggests) an extensible fashion, allowing its use in both wired and wireless LANs. However, it has had its own documented set of problems [12]. Cisco Systems Inc. has developed its own Lightweight Extensible Authentication Protocol (LEAP), information about which can be found in [13]. It is not yet clear if EAP or LEAP solve 802.11's security problems, thus it is wise to combine them with other security techniques.

- **Stealth deployment**

  Wireless equipment is so cheap that people can deploy their own equipment, if it is not provided by their organisation. These deployments may not meet security policies, and those responsible for security may not be aware of them. The network manager would have to find rogue access points and control them, perhaps without the permission of the installer. One mechanism for finding rogue access points is NMAP [10] a free tool for network exploration and security auditing; some access points have IP stacks for management purposes that can, in many cases, be identified.

4. Current best practice

There are three main techniques used to ensure that problematic access networks do not cause problems for sensitive core networks:

1. Treat the wireless network as being a "dirty" network, similar to dial-up. This means the wireless LAN should not bridge Ethernet between a wireless network and a standard desktop or server LAN. If possible, the wireless LAN should be on physically separate equipment. If this is not possible, a special VLAN should be created that has a choke point router where policy can be enforced.
2. Treat the 802.11b clients as requiring the same degree of security as servers and firewalls. An organisational firewall often encourages the "hard outer shell, soft chewy innards" model of security (a phrase coined by Alec Muffett [15]). If the machines behind your firewall can be hacked as easily as the firewall itself, defence in depth is a necessary technique.
3. Use a transport layer that is cryptographically stronger than WEP. If this is not possible, application layer encryption can help to prevent interception and manipulation. Most modern operating systems support a variety of SSH, VPN or IPSEC, all of which can provide end-to-end security regardless of the transport medium. However, there exist man-in-the-middle attacks that attempt to subvert the key-negotiation phase in these protocols [16].

2. Methodology

1. Introduction

We embarked on a series of test runs to establish the best equipment, software, and modus operandi for our survey. In this section, you will find subjective comments on the range of setups we tried.

Total list of equipment used:

1. *Portable computers:* Sony Vaio R600HEK running SuSE 7.3 and Windows 2000; Dell Latitude L400 running FreeBSD 4.6; Mac iceBook 2001 running MacOS X; Handspring Visor Deluxe.
2. *Wireless cards:* Lucent Orinoco Silver card; Buffalo card; SMC EZ Connect Wireless No. SMC2632W; Belkin card; the standard Apple Airport card.
3. *GPS:* Garmin GPS35-HSV; HI-202S; Magellan GPS companion.
4. *Software:* kismet wireless on Linux; NetStumbler on Windows; bsd-airtools 0.2 on FreeBSD; MacStumbler 0.6b on MacOS X.
5. *Transport:* foot; bicycle; car.

2. Hardware

Selection of PCMCIA capture cards.

WiFi cards are primarily distinguished by their chipsets; whatever the brand of the card, inside is silicon produced by one of a small collection of manufacturers. The three chipsets we tested were: Prism 2; Orinoco; Aironet.

Prism 2 can be roughly described as the consumer chipset. It is relatively cheap with an easy-to-use interface to the card internals which enable sniffing. It originates from a company called Intersil. We used two cards utilizing Prism 2 chipsets: an SMC card and a Belkin card.

Some cards that use the Prism 2 chipset are:

Brand name
Addtron AWP-100
Belkin F5D6020
Bromax Freeport
Compaq WL100
D-Link DWL-650
Linksys WPC11
SMC 2632W
Zoom Telephonics Zoomair 4100

Prism 2 cards offer a good compromise between cost and capability. One limitation is that they generally don't connect to external antennae, although some enterprising individuals have made their own [5].

Cards using the Orinoco chipset are one of the most popular on the market. Enabling monitor mode on these requires a special patch to the driver source

code [6] and the firmware must be down- or up-graded to specific values (more details can be found in section 3, Software). We used three cards with the Orinoco chipset: a Lucent Orinoco Silver card, a Buffalo Tech card and the standard Apple Airport installed in iBooks.

Some cards using the Orinoco chipset are:

Brand name
Apple Airport Card
Buffalo Tech
Lucent WaveLAN/Orinoco

The Aironet chipset was developed by Aironet Wireless Communications, who were aquired by Cisco. We used a recent card (350 series), manufactured by Cisco.

Antennae

Unlike Prism 2 cards, Orinoco cards generally come with the ability to add an antenna by means of a proprietary connector. We used a omni-directional antenna for our Buffalo and Lucent cards. Antennae are not strictly necessary when wardriving (we were able to sense many networks without one), but they definitely improve the likelihood of network observation. It is possible to construction one's own antenna [1] [5] if one wants to save money.

Here is an illustration of the antenna used during observation.

Buffalo Antenna used during observation

Closeup of Lucent antenna connector

Laptop Capabilities

Certain Macintosh and Sony computers have built in wireless cards. If your laptop does not, a PCMCIA slot is effectively a must. As GPS units connect to serial or USB ports, one of these is necessary. Long battery life is essential. Our Sony performed poorly in this regard, only ever managing to run on battery power for two hours (despite turning the screen to minimum brightness, minimizing disk accesses/spin-ups and reducing the CPU speed). While warcycling, one of us (DM) found Mac audio support a useful way of informing him of new networks.

Here is an illustration of the computers involved in our survey.

Dell Latitude on top of the car used for wardriving

Dell Latitude closeup with USB GPS unit

Mac iBook 2001 with warcycle

GPS

Many "server class" GPSes come with serial output only. RS-232 serial is increasingly being deprecated in favor of USB. We used a serial GPS with a serial to USB convertor purchased at Maplins and a USB GPS. The main problem with the serial GPS is power consumption. The serial GPS we used was modified to take power input from a car cigarette lighter. When run off 9 volt batteries, they drained in approximately 30 minutes.

Be warned: GPS equipment can produce output as (degrees), (degrees.minutes) or (degrees.minutes.seconds). We noticed this as some of the devices we sniffed initially appeared to be in the middle of Dublin Bay, until we converted all the points to the same (D,M) format.

Here is an illustration of our modified GPS setup.

GPS unit with cigarette lighter attachment and serial port

3. Software

Operating systems and Sniffer applications

Of the packages we tried, the best run on free UNIX operating systems. On Linux, kismetwireless [8] was written with sniffing in mind. For FreeBSD, bsd-airtools provides similar functionality. Windows has NetStumbler, though it does not support features like packet logging and passive sniffing.

Setting up kismet to run correctly is not trivial on Linux: firstly, a patched version of either the Prism 2 or Orinoco wireless card drivers must be obtained

[6] and new modules built that allow the card to be put into RF monitor mode. These modules must be put in a standard place and the PCMCIA system configured to load them in response to card insertion. Secondly, the user application must be configured to talk to the card in the right way, which is application dependent. In the case of the latest release of kismet, writing the card type into the configuration file, and launching the channel hopper when appropriate, suffices. Further details are, generally, available in the application documentation.

Power management (for Advanced Configuration and Power Interface (ACPI) based laptops in particular) can be better on Windows than on free *NIX, so there is a tradeoff between operating time and convenience. Active scanning, because it actually sends out packets, uses more power than passive scanning.

4. Transportation

Car ("wardriving")

To get the most out of wardriving, preparation is essential. Firstly, there must be enough room for the participants and the equipment. In particular, power and communication cables from GPS unit must be routed so as to cause no impediment to the driver.

Secondly, if private roads (for example: those that run through certain business parks and industrial estates) are to be examined, concealment of the equipment may be necessary. This can be achieved by moving wires underneath seats, covering laptops with coats and so on. It is useful to have an OS that will ignore Advanced Power Management (APM) or ACPI events that normally result in a power-down or suspension of the system. Alternatively, sniffing can also be done with PDA-class units such as the HP iPaq, which is extremely easy to conceal; a version of NetStumbler exists that talks to a PCMCIA card attached to an iPaq by a proprietary jacket interface.

In the majority of cases, however, no concealment is necessary. Our best results were obtained with the antenna pointing out of the rear window of the vehicle, affording an unobscured radio view.

Cycle ("warcycling")

If you do not have a USB GPS, cycling becomes practical given the software in use has a speech synthesis or audio feature for announcing new networks. Using headphones that don't interfere with the ability to hear the traffic is essential. Having an audio "heartbeat" is useful, as it is easy to tell if the headphones have become disconnected or the laptop has gone to sleep.

In areas of light traffic, basic GPS can be collected manually using the "waypoints" feature of a hand held GPS unit. We used a Magellan "GPS Companion" for the Handspring Visor, though version 3.0 of the "Nav Companion" software is, in our experience, unreliable.

Other

A certain amount of the survey was conducted on foot, particularly on university

campuses. It was possible to pack a laptop, antenna and GPS unit discreetly. We considered public transport, but did not have time to test the idea; the radio view from the top of a two-deck bus might have revealed more networks.

5. **Cost**

   To engage in a survey like this is not particularly costly, if person-hours are excluded. Many of the resources required would already be available to businesses and individuals. In Ireland, a suitable laptop can be bought for under 1200 Euro, a wireless network card for under 150 Euro, and a GPS unit for under 150 Euro; the software is free.

6. **Location Selection**

   In the course of the survey, we covered more than a thousand kilometers. The emphasis was placed on the city center and industrial areas. Residential areas were investigated less comprehensively, but, for completeness, every road and lane in one residential area, Clontarf, was searched.

7. **Mapping**

   Once GPS latitude and longitude locations are collected they have to be transformed into coordinates suitable for plotting on an image. To produce this transform, we used three fiducial points in Dublin that are easy to locate on maps and satellite photographs.

   Finding freely available electronic maps of Dublin wasn't easy. Some street maps and aerial photographs are available from map-servers on the web. One NASA web page also provides a satellite photograph of Dublin. Shown below are the latitude, longitude and image coordinates for this particular satellite photograph.

Fiducial Point	Latitude	Longitude	x	y
Bull Island Intersection	53.37400	-6.16439	1559	656
Outside US Emb. Phoenix Park	53.35945	-6.32645	1152	501
Merrion Gates	53.31644	-6.20491	1338	831

   From these fiducial points, C code, which can be found in the appendix, calculates the transformation from general points.

   ERA Maptec kindly gave us the best of our maps, which can be found in the next section.

3. Survey Results

1. Pictorial

Our survey results are in two formats: pictorial and tabular. The pictorial results are maps of Dublin with plots of where we have encountered wireless network activity. We have written a program to plot the GPS coordinates of the Wireless Access Points (WAPs) against a rastered (constant) background map. The

program supports a range of features that includes: indicating WAPs with/without WEP; searching for the BSSID of a particular WAP; zooming.

You can find the program here (note; this opens the program in a new window). There are pre-generated images also available.

This, for example, is a satellite map of Dublin. WEPed networks are cyan and unWEPed networks are red. Dublin has a population of approximately 1.25 million. Cloud obscures the tombolo to Howth in the North-East corner. The center of the city is to the south of the main river, the Liffey, with some financial institutions on the north side of the Liffey. There is a ring-road motorway, the M50, around the outskirts of the city, at the northern-most point of which runways from Dublin International Airport can be seen. Industrial estates and business parks account for much of the land adjoining the M50. The clump of networks in the extreme south are in Sandyford's large industrial estate, to which the M50 will ultimately lead.

2. **Tabular**

Summary of results:

Number of access-points, with WEP and GPS

Summary of Results	
Total number of stations	378
Stations with encryption	146 (38.62%)
Stations without encryption	232 (61.38%)
Results with GPS co-ordinates	307 (81.22%)
Results without GPS co-ordinates	71 (18.78%)
Stations with changed SSID	322 (85.19%)
Stations with default SSID	56 (14.81%)
Lowest Latitude (southmost)	53.26997
Highest Latitude (northmost)	53.4135683
Lowest Longitude (westmost)	-6.424875
Highest Longitude (eastmost)	-6.102553

Manufacturer breakdown

Summary by Manufacturer	
agere-lucent	127
cisco-aironet	86
ad-hoc network	44
unknown	31
intel	16
3com	16
apple	14
netgear	7

gemtek	6
breezenet	6
smc	5
d-link	5
addtron	4
nokia	4
compaq	3
acer	1
enterasys	1
adv multimedia	1
linksys	1
	378

Default SSID breakdown

Summary by Default SSID	
(null)	12
101	9
WaveLAN Network	8
tsunami	7
WLAN	7
Apple Network	5
any	3
3Com	3
Wireless	2
	56

Spectrum (channel usage) breakdown

Summary by observed channel	
1	83
10	68
11	47
7	40
3	39
6	32
0	30
13	16
2	7
4	6
8	5

5	3
12	1
9	1
	378

Full dump of MAC address and location

The manufacturer categorisation is done by mapping MAC addresses to manufacturers. We used several publicly available resources to do this, including the Wisconsin 2600 list [9] and the IEEE list of all known MAC address mappings.

4. Conclusions

1. Deployment of 802.11

802.11 has been extensively deployed throughout Dublin. The city centre and business parks where hi-tech companies operate show high penetration, as do the universities and other third-level colleges. No secondary schools were identified as having 802.11 deployments, though this may be due to the time of year at which the survey was conducted.

Residential areas have, in general, sparse 802.11 activity, though there are signs of deployment. Clontarf, on the coast to the north-east, is a residential suburb that was surveyed comprehensively by bicycle. About 10 access points were found; half seem to be residential, with the other half used by home offices or small businesses.

From the collected network names, which we will not publish, although they are easily obtained, and the locations of observed networks, it is clear that many different types of organization have deployed 802.11 networks, including financial, telecoms, computer, engineering, educational, graphic design, local government/public sector, residential and hobbyist. There is obvious deployment of 802.11 networks in its expected function as a computer LAN, but also as a long-distance point-to-point link. One node claimed to be part of the Irish WAN network [17] which is a community network access project.

We found over 150 distinct network names (SSIDs). We broke these down into several catagories to see how people are naming their networks. The most common naming scheme was to name the network after the organisation or a subdivision. A smaller number were named after individuals.

Most of the remainder are named with default names, generic networking terms or miscellaneous english words. Interestingly, a few are named either with random strings (produced by hitting a keyboard?) or by performing transformations on words such as Apple -> 4pp1e. This suggests some people consider the network name to be a password; it's not, and should not be viewed as having the same security semantics. In particular you should **never** re-use a password from another system for either an 802.11 SSID or WEP key.

Some point-to-point links were named by the location of their end-points. A

small number were named using traditional identification schemes such as DNS name or telephone number. Some were named according to an apparently random hex digit scheme; this might be due to confusion with WEP key configuration or some as yet unknown naming scheme.

Below is a table illustrating the percentage breakdown, together with some (invented) examples.

SSID Category	%	Examples
Organisation (external)	29%	Enigma Wavelan
Default/"network"	16%	tsunami, TempLan, network
Probably organisational	11%	TFX
Misc	11%	lion, paris, voyager
Organisational (internal)	7%	Faculty of Science
Random hex digits	6%	0A458C
Owner	4%	bill's airport, theodore
Random/Password Like	4%	t3l3c0, dsaugdiu
Location	3%	dublin office, leesonstreet
Home	3%	home wlan, my airport
Point to Point	2%	dawson to oconnell
Domains	2%	maths.tcd.ie
Function	1%	laptop network
Phone numbers	1%	UIS353162X5678

Wireless networks seem to have fixed positions and are usually on all the time. Along one author's route to work, 12 have been observed. One has appeared since the survey began, one has disappeared, and 3 are not observed on every survey. Our experience is that it is easier to locate business networks during the working day and easier to observe residential networks in the evening; this should be expected with the passive sniffing technique, as it relies on packets being generated to find networks.

We also found evidence of commoditisation of 802.11. Networks were detected in a restaurant with a computerised order management system and also near some large retail units.

One observed phenomenon of ad-hoc networks was the apparent incorrect generation of BSSIDs. For IBSS or ad-hoc networks the BSSID should be generated by taking 46 random bits r and constructing the address:

```
rrrr rrug:rrrr rrrr:rrrr rrrr:rrrr rrrr:rrrr rrrr:rrrr rrrr
```

where u is the Universal/Local bit set to 1 to indicate a local address and g, the Individual/Group bit, is set to 0 to indicate individual.

Some ad-hoc network equipment sets the top byte to 02 then correctly sets the rest randomly. These ad-hoc networks (while having no association) seem to generate a new BSSID every 10 seconds. Approximately 36 of the BSSIDs

collected may be attributable to two devices using this scheme.

2. **Spectrum usage**

   TV, commercial radio, mobile phone and even amateur radio are examples of wireless networks that now enjoy widespread deployment. The operation of all these networks is protected by licensing of the relevant pieces of spectrum. This protection is both legal (e.g. the shutting down of "pirate" radio stations) and social (e.g. radio hams complaining about sloppy operation of equipment).

   WiFi is in a different situation. The spectrum is free for general use, subject to constraints on transmission power. There are many possible sources of interference to which a 802.11b network is subject, including portable phone handsets, wireless keyboards and mice, wireless digital cameras, bluetooth devices, microwave ovens, and other 802.11b networks.

   Anecdotal evidence suggests that unintentional overlap between 802.11b networks is common; in network administrator circles, there are often stories of laptops associating to neighboring companies' access points.

   Several companies are considering large public commercial deployments of 802.11. By the time these deployments move out of the labs and on to the streets, will there be space in the spectrum for them? It remains to be seen if the mutual good nature of organisations "sharing the Ether" will be sufficient to resolve any disputes that arise; it is likely that a mutual cooperation body or a some kind of formal regulation will be required.

3. **Deployment of WEP**

   American surveys typically report WEP being used on 20-25% of networks. Of the networks we found, 39% had WEP deployed and some others had application encryption. It is possible that reports about the insecurity of wireless networks have had an impact on network managers and their approach to wireless security.

   Though it is weak, WEP should probably be on unless there is a reason not to have it on. For example, on a student LAN in a University or "public" wireless hot spot, there is little advantage in using WEP; the secrecy of the WEP key cannot be assumed, because of the large number of people who know it.

   On residential networks and small business networks, WEP can be enabled without the WEP key being divulged to large numbers of users. On point-to-point links, WEP deployment is trivial.

5. Acknowledgments

We wish to thank the following people who helped in various ways during the survey: Ian Dowse of Corvil Networks, Nick Hilliard of Network Ability Limited, Lean Ni Chuilleanain of the RIA, John Lewis of CNRI, Wayne Sullivan of CNRI, and John Walsh of the Dublin Institute for Advanced Studies. In particular, we wish to thank ERA Maptec for providing us with satellite imagery.

6. Contact Details

Niall Murphy, e-mail: niallm-web@enigma.ie

David Malone and Ken Duffy, e-mail:{david.malone, ken.duffy}@cnri.dit.ie

7. References

Note: these will open new windows.

1. Homebrew antenna shootout
2. Dsniff: Tools for network auditing and penetration testing
3. WEP flaws
4. 802.11 Wireless Networks (The Definitive Guide), Matthew S. Gast, O'Reilly & Associates (2002)
5. Homemade antenna
6. Orinoco driver source code patch
7. WEP studies
8. Kismetwireless
9. Wi2600 vendor list
10. NMAP: a tool for network exploration and security auditing
11. RFC
12. EAP problems
13. LEAP white paper
14. Security Engineering, Ross Anderson, Wiley & Sons (2001)
15. Alec Muffet's home page
16. Airjack
17. irishwan.org

8. Appendix

1. C code for calculating coordinate transforms

To calculate the transform requires matrix algebra. If g0, g1, g2 are vectors of the GPS coordinates and p0, p1, p3 are the corresponding coordinates in the satelite image, then calculate:

```
tt[x_,y_] = p0 + Transpose[{p1-p0 , p2-p0}].Inverse[Transpose[{g1-g0, g
                ({x,y}-g0)
```

The program that follows performs this calculation:

```
int main(void) {
        double x[3] = {53.37400, 53.35945, 53.31644};
        double y[3] = {-6.16439, -6.32645, -6.20491};
        double xp[3] = {1559, 1152, 1338};
        double yp[3] = { 656,  501,  831};
        double M1[2][2], M2[2][2], M3[2][2];
        double det;

        M1[0][0] = xp[1] - xp[0];
        M1[1][0] = yp[1] - yp[0];
        M1[0][1] = xp[2] - xp[0];
        M1[1][1] = yp[2] - yp[0];
```

```
        M2[0][0] = y[2] - y[0];
        M2[1][0] = -(y[1] - y[0]);
        M2[0][1] = -(x[2] - x[0]);
        M2[1][1] = x[1] - x[0];
        det = M2[0][0]*M2[1][1] - M2[1][0]*M2[0][1];
        M2[0][0] /= det;
        M2[1][0] /= det;
        M2[0][1] /= det;
        M2[1][1] /= det;

        M3[0][0] = M1[0][0]*M2[0][0] + M1[0][1]*M2[1][0];
        M3[0][1] = M1[0][0]*M2[0][1] + M1[0][1]*M2[1][1];
        M3[1][0] = M1[1][0]*M2[0][0] + M1[1][1]*M2[1][0];
        M3[1][1] = M1[1][0]*M2[0][1] + M1[1][1]*M2[1][1];

        printf("xp = %f * x + %f * y + %f\n", M3[0][0], M3[0][1],
                 M3[0][0]*(-x[0]) + M3[0][1]*(-y[0]) + xp[0]);
        printf("yp = %f * x + %f * y + %f\n", M3[1][0], M3[1][1],
                 M3[1][0]*(-x[0]) + M3[1][1]*(-y[0]) + xp[0]);

        return(0);
}
```