

The Accuracy, Secrecy and Testing of the Nedap/Powervote Electronic Voting System

Submitted to the Commission on Electronic Voting by
Irish Citizens for Trustworthy Evoting

WWW: <http://evoting.cs.may.ie/>

25th March 2004

Abstract

The Commission on Electronic Voting has requested submissions regarding the accuracy and secrecy of the Nedap/Powervote system. ICTE hereby submits that the accuracy and secrecy of the system are questionable. ICTE further submits that the accuracy of the system is impossible to verify and thus does not sufficiently safeguard our right as citizens, under Article 6 of Bunreacht na hÉireann, to designate the rulers of the State.

Electronic voting has the potential to offer several benefits, including reducing the number of mistakenly spoiled ballots and making voting more accessible. ICTE does not oppose the introduction of electronic voting, and would welcome a trustworthy system capable of bringing these benefits to our polls. However, we also believe that the dangers posed by the system proposed for use on June 11th are so serious that it is not suitable for use in Irish elections. It is in that context that we make this submission.

The chosen Nedap/Powervote electronic voting system has a fundamental design flaw: it has no mechanism to verify that votes are recorded accurately in the practical setting of an election. Consequently, results obtained from the system cannot be said to be accurate. Other flaws in the system are also identified in this submission.

Contents

1	Introduction	3
1.1	Protecting Democracy	3
1.2	Voter-Verified Audit Trail	4
1.3	Additional Concerns	5
1.4	Structure of this paper	5
2	Accuracy	6
2.1	The ballot record is not voter-verified	7
2.1.1	Voting needs auditing	7
2.1.2	Auditing secret votes requires voter verification	7
2.1.3	Elections have always required voter verification	7
2.1.4	The proposed system lacks voter verification	8
2.1.5	Analogy: Man-behind-the-curtain voting	9
2.2	The ballot record is not accurate without voter verification	9
2.2.1	All software is vulnerable to programming error	9
2.2.2	All hardware is vulnerable to malfunction	11
2.2.3	Voting machines are vulnerable to tampering	12
2.2.4	Inspection cannot discover well-hidden tampering	12
2.2.5	A corrupt toolchain can hide tampering from reviewers	13
2.2.6	The backup module does not offer fault-tolerance	13
2.2.7	Significant real-world risks are untested in pilots	14
2.2.8	Some funded organisations pose a direct threat to accuracy	14
2.2.9	The voting machine seals are vulnerable	15
2.3	The ballot record can be changed during counting	15
2.4	The counting is not necessarily correct	16
2.4.1	Lack of supervision opens counting to new risks	16
2.4.2	Errors in counting PCs are likely	16
2.4.3	Tampering with counting PCs is possible	17
2.5	Inadequate consultation	17

3	Secrecy	19
3.1	Votes are not stored randomly as required	19
3.2	Voters cannot abstain in secret	20
3.3	Voting machines can be programmed to violate secrecy	20
4	Testing	22
4.1	Review of consultants' reports	22
4.2	Lack of adequate end-to-end testing	23
4.3	Inadequate security criteria	24
4.4	Unreviewed machine code	24
5	Conclusion	26
	Bibliography	27
	Appendix A: Contributors	29

1 Introduction

“Constitutional rights are declared not alone because of bitter memories of the past but no less because of the improbable, but not to be overlooked, perils of the future.”

Chief Justice Ó Dálaigh, speaking for the Supreme Court, *McMahon v. Attorney General* [1972] IR 69.

Irish Citizens for Trustworthy Evoting (ICTE) is an alliance of citizens, brought together by serious and legitimate reservations over the introduction of electronic voting in its proposed form. ICTE was established formally in May 2003 by Ms. Margaret McGaley but it has grown quickly and includes some of those that had previously been lobbying and raising their concerns in isolation. Currently its membership comprises over one hundred people, including technical and legal experts.

This document is the product of cooperation and open discussion between the members of the group via a public electronic forum, to which any person may subscribe and contribute. As such, it represents the common consensus amongst our membership.

1.1 Protecting Democracy

The accuracy of the proposed electronic voting system must be seen in the context of elections and national votes in general. These are the very foundations of a democratic society. Citizens expect their votes to be confidential, to have equal status and to be protected from alteration or loss. Any voting process, no matter how it is implemented, must have sufficient measures in place to maintain citizens’ expectations of **accuracy** and **secrecy**.

It should be recognised from the outset that some may seek to pervert the course of the democratic process by attempting voting fraud. These illegal and unjust efforts may be made by individuals, groups or potentially by well-funded organisations. It is not necessary to show that tampering is likely in order to justify acting to protect against it. Elections are manifestly adversarial events in which there are clear motives to affect the result.

With the introduction of electronic technology into elections there also comes an entirely new and unfamiliar risk model. It is not unusual for electronic systems to be attacked a relatively long period before any benefit is taken from such an attack. Frequently, attacks on electronic systems are practically undetectable, and many such attacks require minimal expertise.

Excluding deliberate attacks, electronic systems are inherently prone to random, unavoidable and naturally occurring causes of error. Inadvertent failures of hardware and software have occurred in voting machines in other countries. This has been most widely documented in US voting machines, but has also been seen in Belgium, where a single-event upset (most likely a cosmic ray) caused a 4,096-vote error in declared results. A similar naturally occurring and unavoidable error could occur with the Powervote counting PCs, and we may not notice it unless it is large enough to be absurd.

In Fairfax County, Virginia in November 2003 [Cho03], direct recording electronic (DRE) voting machines were seen to change the voter’s choice on the screen from one candidate to an-

other, in favour of the same candidate in each case. Some voters reported this and a machine was tested and found to do this with about one per cent of votes. This error could be corrected by the voter selecting the intended candidate again. Since the voters could see the vote being changed on the screen, it is likely that most of them corrected it, but that some did not notice the change. Since there was no independent, voter-verified record of the votes, it is impossible to tell what the result would have been if all voters' intentions had been recorded accurately.

Although this fault was visible to the voter, it is equally possible that a software bug or hardware fault could cause votes to be recorded wrongly while being displayed correctly. In that case, unless the result was implausible, it is very unlikely that this error would be discovered, since there would be no reason to suspect it.

This case also demonstrates that a fault can cause a machine to throw a small proportion of votes to a particular candidate, thereby potentially altering the result in a close election, without raising suspicion of an error.

While this case relates to a different model of voting machine to that used in Ireland, the nature of the risk is the same, since both are direct recording electronic machines with no voter-verified audit trail.

1.2 Voter-Verified Audit Trail

Central to our concerns about the system is the absence of a voter-verified audit trail (VVAT), also called a voter-verified paper audit trail (VVPAT) or a voter-verified paper ballot (VVPB). A VVAT is the only practical means by which voters may be assured that their vote has been recorded correctly. Without it, the accuracy of an electronic voting system cannot be verified in any way that is independent of the system itself.

By “voter verification” we mean the process by which each voter personally ensures that the vote recorded on his/her behalf is identical to the vote actually cast. The only way voters can verify that their votes have been recorded accurately is by observing that recording themselves. This implies that verification has not occurred if the voter is only told what has been recorded, by another human or by a device. If the Nedap/Powervote system were modified to include a VVAT, voters would see their own votes on paper, as they did in the all-paper system, and would not have to trust in the assurances of any person or device.

By “voter-verified audit trail” (or “VVAT”) we mean a physical token (usually paper) which: (a) is verified by the voter in each case; (b) is subsequently so handled that tampering is impractical; and (c) is the final and authoritative record of the vote in the event of disputes.

Although we will deal with many particulars of the chosen system, it is important to note that the lack of a VVAT alone is a critical design flaw and is sufficient to render any such electronic voting system untrustworthy, and its accuracy unknowable. A VVAT is a simple, independent record of our votes — verified by us as voters — and is the only method which allows for verification of the end result.

VVAT is considered essential by independent computer security and electronic voting experts such as Bruce Schneier [Sch00] and Dr. Rebecca Mercuri [Mer01]. Organisations, including

the USACM¹ [ACM] and the electronic voting panel of the ICS² [Soc04] have issued statements declaring that VVAT is necessary for trustworthy electronic voting. Section 2 explains this topic and its relevance in more detail.

1.3 Additional Concerns

When testing a car, mechanics would never examine individual parts without testing the car as a whole. Such an approach is inherently prone to error³, but it is analogous to the manner in which the Nedap/Powervote electronic voting system has been tested.

While concerns over improper testing are incidental to the fatal lack of a VVAT, we are nonetheless submitting additional evidence in this regard. This is done to highlight relevant, realistic weaknesses which may affect electronic voting systems in general and, more specifically, the chosen system.

1.4 Structure of this paper

Section 2 discusses requirements for accuracy of voting systems and shows that the Nedap/Powervote system does not meet those requirements.

Section 3 discusses three ways in which the Nedap/Powervote system violates the ballot secrecy requirement.

Section 4 examines the testing to which the Nedap/Powervote system has been subjected, and shows that it has been inadequate.

Finally, section 5 discusses the importance of security and of anticipated fraud, and in particular the relevance of these matters to the Commission's task.

¹U.S. Public Policy Committee of the Association for Computing Machinery — an international body for computer professionals

²Irish Computer Society — the national body for computer professionals

³Fred Brooks, in his seminal work [Bro75, chapter 13], said “The most pernicious and subtle bugs are system bugs arising from mismatched assumptions made by the authors of various components.”

2 Accuracy

The proposed e-voting system is “accurate” if and only if in every election the results announced by the system correctly reflect the cast votes.

There are three necessary conditions for establishing the correctness of any such result:

1. The record of ballots always correctly reflects the cast votes; that is, either:
 - (a) the ballot record is voter-verified; or
 - (b) the ballot record is necessarily accurate without voter verification;
2. The record of ballots cannot be changed until after the election is over; and
3. The counting process always correctly computes the election result from the recorded ballots.

A defect in any one of these three conditions must cause a claim of accuracy to fail. In this section, we show that all three conditions are not met:

1.
 - (a) Subsection 2.1 shows that the proposed system does not support voter verification;
 - (b) Subsection 2.2 shows that any ballot record made in a real-world election, without voter verification, is vulnerable to error, malfunction, and tampering, and therefore cannot be reasonably expected to always be correct;
2. Subsection 2.3 shows that the ballot record may be changed, either in the ballot modules or in the count PCs internal database (whether due to error, malfunction or tampering); and
3. Subsection 2.4 shows that the computed election result is not reasonably likely to always be correct.

It is our submission that the design of the Nedap/Powervote system, proposed for use on June 11th, is critically flawed in that its accuracy is not established at present, is not secure against fraud in the future, and is categorically not open to verification. Put simply, there are no means by which we can be reasonably confident that the results, as announced, will accurately reflect the intentions of the electorate. Particularly in closely fought constituencies, it will be impossible to determine that a result is free from inaccuracy either through error or malice.

Even if the proposed system as designed is considered to be secure, it is impossible to practically verify that the system used on a polling day is the same as the Nedap/Powervote system which has undergone testing. **It is meaningless to say that the system has been tested when it is not possible to be assured that the tested system is the one in use.**

2.1 The ballot record is not voter-verified

In this section we show that voting is an important system which needs auditing; that in the context of voting, auditing must mean voter verification; that historically we have had voter verification; and that with the proposed system we don't have voter verification. Finally, we briefly illustrate why a lack of voter verification is unacceptable.

2.1.1 Voting needs auditing

Almost all important electronic systems have been attacked in attempts to defraud them. Successful attempts are usually only detected when auditing indicates a discrepancy (such as when a bank customer checks his/her account statement). To support this auditing, detailed audit records are kept of every significant transaction. No electronic commerce system would be deployed without an auditing capability; fraud in the real world is too frequent for any one business to absorb the risk.

Regardless of the number of tests, reports and audits performed on the chosen system, the lack of an independently verified record of our vote will render the system prone to the same problems inherent in any other unaudited electronic system. As USACM said in [ACM], "computers are inherently subject to programming error, equipment malfunction, and malicious tampering."

2.1.2 Auditing secret votes requires voter verification

Voting systems present an unusual audit challenge in the form of ballot secrecy. A full set of audit records cannot be kept in the usual manner, as it would facilitate the prohibited act of correlating voter identities with ballots. For these systems, a different approach to auditing must be found.

The answer was identified in 1992 by Dr. Rebecca Mercuri [Mer92], then a Research Fellow at the University of Pennsylvania. Because only the voter is permitted to know the details of his/her vote, that voter must "audit" the record of the vote, and the fact of this audit must be retained in undeniable form. This is a generalised procedure; although the statement of it was new twelve years ago, the procedure itself was not new, and has been followed in practical systems since the paper ballot was introduced.

2.1.3 Elections have always required voter verification

It has long been an established requirement of an election that it be voter-verified. The reasoning behind this is clear: it is important that all sides be assured that the result is accurate, that tampering has not taken place and that human error has not occurred. It would not be acceptable to have the procedures of a manual count hidden behind closed doors and the result announced by decree.

Voter verification was a natural and integral part of the paper-based system used here for over 100 years. One could place one's own vote in a solid metal ballot box, watch the ballot box being

sealed, confirm that the seal was unbroken when opened at the count centre and see the same vote being counted, all with one's own eyes. In addition to this, an independent preliminary count of the first preferences of our votes was present in the form of the traditional tally. Should a tally and the official count of first preferences not match, one could be sure that the level of scrutiny would be raised.

The correct recording of one's vote could be verified with one's own eyes. All that was necessary was the knowledge that pencil lead does not fade overnight. The correct counting of one's vote was safeguarded by the knowledge that in the event of a close count recounts would be ordered, with ever-increasing levels of scrutiny until all sides were assured of the veracity of the result.

The present paper ballot system incorporates a voter verified audit trail. The ballot paper issued to each voter is in the exclusive and uninterrupted control of the voter from the time it is franked to the time it is dropped into the ballot box, and it is subsequently secured by well-established procedural controls. The voter is justifiably certain that the ballot paper contains only the marks he/she has made on it. Although the ballot paper is principally used for the initial counting, it is also the final authoritative record of the voter's preferences.

2.1.4 The proposed system lacks voter verification

With the introduction of electronic voting, this is no longer the case. Electronic systems inherently deal with the intangible: it is not possible to see an electronic record with one's own eyes. Just as it is impossible to distinguish between two compact discs looking only at the reflective side, it is impossible to make out the sub-microscopic electronic charges storing our votes on a Nedap ballot module. Thus electronically recorded votes are not open to human verification, because the vote recording and counting mechanisms — the voting machines and software — are not open to human verification.

Electronic hardware deals with information at a sub-microscopic level. When electronic hardware records information, the physical record is minute and not observable outside of an expensive laboratory environment. In a completely electronic voting system, the vote is recorded by an electronic device, out of sight of the voter. There is no direct connection between the voting machine's display and its electronic vote records. It is perfectly possible that a computer — either by accident or malicious design — could show one vote to the voter and record a completely different vote.

Therefore, to the voter and election officials:

- a Nedap voting machine that is recording our votes correctly may appear completely identical to, and indistinguishable from, a Nedap voting machine that is recording our votes incorrectly.
- a Nedap ballot module with our votes recorded correctly may appear completely identical to, and indistinguishable from, a Nedap ballot module with our votes recorded incorrectly.
- Powervote counting software which counts our votes correctly may appear completely identical to, and indistinguishable from, Powervote counting software which counts our votes incorrectly.

The implication of this is severe; without VVAT there is simply no means by which any person may be confident that his/her vote has been recorded and counted correctly.

2.1.5 Analogy: Man-behind-the-curtain voting

Consider this hypothetical balloting system: each voter, having been authorised to vote, is directed to whisper his/her vote to a voting official hidden behind a curtain. The official records the voter's intentions in some way out of the sight of the voter. Secrecy is assured by the fact that the official does not know the identity of the voter. Accuracy is allegedly assured by the occasional practice of subjecting the official to tests to see if he/she correctly records known test votes. In such a system, lacking voter verification, voters would be uncertain that the official behind the curtain was truly recording votes honestly and accurately. In particular, the official may be predisposed to influence, either outside the polling place or in response to something said by a voter. Such a system would clearly be unacceptable.

The proposed electronic voting system, which relies on a computer of unknown programming rather than on an official of unknown probity, is surprisingly comparable. To a layman not well versed in the design of modern electronics, it may seem that experts can analyse a computer to establish its reliability and accuracy, but as section 2.2.4 shows, this is not the case.

2.2 The ballot record is not accurate without voter verification

In this subsection, we show that software is intrinsically vulnerable to programming error and malfunction; that computer hardware is vulnerable to the same risks in a different way; that both hardware and software are vulnerable to tampering; that inspection of computers to verify them is an unrealistic hope; that inspection of source code is not adequate; that the promised backup module is a placebo; that the apparent success of the pilots is no guide to the future performance of the system; that there is a real risk of attack on election integrity by funded organisations; and that the use of seals on each voting machine is a particular weakness.

2.2.1 All software is vulnerable to programming error

The term “software” refers to a set of coded instructions for a computer. As with any human endeavour, producing computer software is prone to error. It also presents a set of unique challenges which are unfamiliar to non-specialists. Software writers create documents (known as *source code*) which are precise, readable specifications that are mechanically translated into the precise but unreadable *machine code* that the computer's processor can execute directly.

Although well-defined and easily readable to a machine, source code is not always immediately obvious even to expert programmers. For example, the following two snippets of “C” source code do entirely different things:

```
if (foo == bar && apple | orange )
```

```
if (foo = bar & apple || orange )
```

Although side-by-side the differences are obvious, when embedded inside hundreds of lines of source code these kind of differences are notoriously difficult to spot. All programming languages including the languages used by the Nedap/Powervote software, C and Object Pascal, display these characteristics.

For this reason, it is generally essential that source code be written as legibly as possible. Important names within the source code such as “variable” and “function” or “procedure” names should be named naturally, *i. e.* `TotalCount` instead of `TtlCnt`. Furthermore, source code should be accompanied by verbose, explanatory commentary.

In spite of simple measures such as verbose commentary, software suffers a large amount of “bugs”, defects with the source code that cause the wrong set of actions, usually only in a particular set of circumstances. Estimates differ on the average rate of defects per line, and it depends on the exact coding style used, but generally 1 defect per 1,000 lines of code would be considered a very good defect rate.

NASA, for example, regularly achieves this and better defect rates in its software; but only after enormous effort — see [Fis96]. The onboard shuttle launch software, representing 420,000 lines of source code, is produced by 260 people and has been subject to intensive audit, screening and control over a period of decades. In spite of this, the software is still not afforded a position of trust and a second software implementation is on standby should it fail.

All responsible software engineers will admit that bugs are an inevitable consequence of programming. It is irresponsible, unprofessional and misleading to claim that any software is “100% Accurate” and “Safe from Hacking” as Powervote have [Pow03]. A large software project without bugs is unheard of. The Nedap/Powervote software combined represents over 220,000 lines of source code, and much of it has been developed by just two people at Groenendaal Bureau. In every review of the software to date, bugs have been found. We submit that beyond any reasonable doubt, **the Nedap/Powervote software does contain bugs.**

Recognising that the primary cause of error in software is the programmer, there is an important principle in good programming called the KISS principle, which humorously expands to “Keep It Simple, Stupid.” Experience shows that the less complicated you design your software to be, and the fewer the number of lines of source code you write, the fewer mistakes you are likely to make. This is an extremely effective means by which software can be protected from unintentional defect.

This is a pervasive and growing concern in software: as libraries, interfaces, and tools become more complicated, they become less understood and less controllable. When everything works, rich programming environments can be very productive, but when they fail, there is little recourse. Indeed, we may not even realize that something is wrong if the problems involve performance or subtle logic errors.

Kernighan and Pike, [KP99, s. 3.9]

The Nedap/Powervote software design is over-complicated and hence does not follow the KISS principle:

- The design of the Powervote software is **monolithic**. Rather than segment the requirements of the software into well-defined and separate software programs, Powervote have placed all of the functionality in one enormous software program. The Powervote counting software is not only responsible for the counting of our votes, but the printing of ballot papers, the output of tabulated results and so on. Any security-minded software engineer will confirm the seriousness of such a basic design flaw.

It is possible to write a software implementation of the Irish counting system in, at most, 1,000 lines of source code and people have done so [Mal98]. By embedding the vote-counting software elements in a large application of over 200,000 lines of code long, Powervote have opened the design to the possibility of error or subterfuge in an unrelated part of the same application (which, potentially, has not been subject to the same level of scrutiny and audit) causing an inaccurate result to be reported.

- Computer science offers many programming languages to a source code writer. We have already mentioned two — C and Object Pascal — that are used by the Nedap/Powervote system. Each language has its own characteristics, anachronisms, areas of suitability and so on. Within computer science and the computing profession it is almost **unheard of** for Object Pascal to be considered suitable for use in a safety critical system.

Borland Delphi, the Object Pascal programming environment used by Groenendaal Bureau to develop IES, is a “Rapid Application Development Interactive Development Environment”. Its primary aim is to facilitate rapid development of “graphical user interface”-based applications. Although in theory any program may be written correctly in any programming language, there is simply no community, body of experience or corpus of knowledge built up around using Object Pascal for safety-critical systems.

- By using Microsoft Access for intermediately storing votes, Powervote have introduced unnecessary complexity merely for their own convenience. This is not indicative of professional security-minded safety critical software design.

2.2.2 All hardware is vulnerable to malfunction

Electronic hardware is based on fundamental components such as transistors, capacitors, inductors and logic gates. In the context of an electronic voting system and accuracy there are some aspects of electronic hardware which are particularly relevant:

- Electronic hardware is typically made up of millions of components and is inherently prone to manufacturing errors — which in many cases go undetected until actual production use.
- Electronic hardware is inherently prone to naturally occurring random causes of error. Electrical power fluctuations and natural phenomena such as highly charged cosmic rays can cause “spontaneous bit inversion”, a process by which Electronic Hardware may suddenly reverse the value of a particular record. Although it is sometimes possible to detect and protect against this, it is not always possible. There are many electronic components

of the system chosen for use in Ireland which have absolutely no protection against this.⁴

The most serious threat to accuracy is the simple likelihood of programming or computer error. The counting software and PCs in particular have not been designed with security or best practice in mind and it is entirely possible that an error may cause an inaccurate result to be reported.

2.2.3 Voting machines are vulnerable to tampering

Computers follow instructions one by one, each one drawn from an *instruction store* and carried out by the circuitry of the *processor core*. The circuitry is laid out on layers of semiconductor (*e. g.* silicon) which are protected inside a block of plastic or ceramic called a *package*. The package, which has visible metal pins to carry signals to and from the circuitry, is often referred to as a *chip*.⁵

Any program can be rendered as microcircuitry and added to the design of a chip. For instance, a chip in the voting machine may be redesigned to monitor the operation of the machine and detect when buttons are pressed and votes are cast. The chip could behave normally until a certain condition is met, and then start interfering with the contents of votes being recorded. The “certain condition” could be a particular long sequence of button-presses by an early voter, or it could be a simple timer, or any other computable condition. This would, if undetected, allow a voter to effectively reprogram a voting machine under cover of casting a vote. Commercial “microfabs” routinely manufacture chips and packages to customer specification; it would be easy for a computer hardware engineer to create such chips, and impossible to tell them from the original by any means short of physically breaking open the packages and examining the chips’ layers under a microscope.

However, it is not necessary for an attacker to go to such expense, since the voting machines’ software is easier and cheaper to change. The software is stored on EPROM chips (a form of reusable data storage) inside the voting machines. Consumer-grade devices are used to reprogram such chips. Although a laboratory examination of the chips would reveal a change in their contents, it is important to note that the change would *not* be noticed by a returning officer conducting an inspection of the software version numbers and checksums.⁶

2.2.4 Inspection cannot discover well-hidden tampering

If the voting machines or counting PCs contain uncertain hardware or software, then it is impossible to have absolute confidence that they will operate correctly and honestly. The so-called

⁴The voting machine design includes measures to reduce the rate of these errors, but the counting PCs are utterly unprotected.

⁵Technically, “chip” refers to the tiny slab of semiconductor material inside the package.

⁶See the review of the PTB report in subsection 4.

primary inspection⁷ by voters will not indicate any discrepancy because the voting machine will display the voter's vote correctly even when programmed to record a different vote. The secondary inspection by presiding and returning officers will not indicate any problem because the machine will still present the expected checksums and version numbers. A tertiary (forensic lab) inspection would detect software changes but not hardware changes. Only microscopic examination could conclusively prove that the hardware had not been tampered with, and the examination would necessarily destroy the unit. It is therefore practically impossible to conclude in respect of any modern computer (including the voting machines and count PCs) that it has not been tampered with during its months of storage or before initial delivery. Where the risk of tampering by funded organisations is significant, as it is in elections and electronic commerce alike, there is no practical way to allay the risk by inspecting the machines.

2.2.5 A corrupt toolchain can hide tampering from reviewers

Only when one has personally constructed and programmed a computing system from raw material and first principles can one trust its behaviour, setting aside for a moment naturally occurring random errors. In the case of unaudited electronic voting, a voter must have faith in the abilities and probity of a wide range of component manufacturers, vendors, consultants and officials.

For example in the case of the Powervote IES software to be used in Irish polls: not only must a voter trust the review of the IES source code, but he/she must also trust the probity and behaviour of Borland Delphi (the development environment), the Borland Libraries, the Borland Compiler, the Microsoft Jet database library, Microsoft Access, Microsoft Windows libraries, the Microsoft Windows Operating System, the programming/reader unit (PRU), the counting machine hardware and all persons who have had access at any time to these components.

With suitable auditing, any problem is recoverable because computer 'errors' can be detected. The only suitable auditing for a secret ballot system is voter verification. Without voter verification, the system cannot be trusted to do what it is supposed to do; rather, it will do exactly what it is programmed to do, whatever that is.

2.2.6 The backup module does not offer fault-tolerance

Nedap backup modules, as is the case with any items of electronic storage equipment, are prone to failure. Although a backup module is in place within the Nedap voting machine, its usefulness for the purposes of redundancy and resilience is somewhat limited.

Rather than storing votes in both modules at the time of voting, which would provide resilience in the event of a module failure during recording, the backup module is synchronised with the primary module only after the close of polls. Any recording errors which have occurred during the polling day in the primary module will be copied onto the backup, thus negating any resilience in the face of live recording error.

⁷This terminology is borrowed from the anti-counterfeiting industry.

Additionally, due to the procedures in place, it is unlikely that a failed backup procedure to synchronise the two modules would be noticed until such time that a returning officer attempted to use a backup module. However, the scenario in which the backup module is utilised is also the scenario in which the primary module has already failed.

The backup module provides no meaningful fault-tolerance with regard to recording errors, and is useful merely for protecting against loss or damage to the primary module in transit (a process which takes far less time than the length of a polling period).

2.2.7 Significant real-world risks are untested in pilots

The electronic voting system cannot be evaluated based solely on its performance in laboratory tests, pilot projects, or other benign environments. It will be used to allocate political power, a function which has consistently attracted fraud throughout history. A serious attacker would be prudent to wait until the system is adopted and accepted by a majority of the people before interfering. The system is planned for use in actual elections, when the results will be broadly accepted by the public so long as they are reasonably close to pollsters' published predictions. The accuracy of the system therefore cannot be divorced from the security threats to the system.

In such a realistic environment, it is not enough to put the burden of proof on the plaintiff challenging the accuracy of the result. The tradition has been to show the correctness of the result by punctilious and exhaustive transparency in the procedures of the election. It is still important to retain this tradition, all the more so if the result will depend on the correct operation of inscrutable computing machines.

To put it bluntly, there is ample opportunity for a well-placed person to reprogram voting machines or counting PCs, and to cover his/her (electronic) tracks well enough to escape detection by election officials and candidates. Naïve attacks will be detected, but moderately sophisticated attacks will not, as the techniques necessary for detecting them are too expensive to be practical. The result of an election subject to this unstoppable and practically undetectable risk cannot be called accurate except with a great deal of optimism.

2.2.8 Some funded organisations pose a direct threat to accuracy

An election is an event of international importance, and however unlikely it may seem, one must take into consideration the possibility that well-funded organisations — including foreign governments and terrorist groups as well as organised crime — may have an interest in affecting the outcome. It is not hard to envisage it being in a foreign government's security interests that a particular party not do well in an Irish election, for example.

One need only be reminded of various national security agency bugging and tampering in international UN offices, or the revelation that the electronic encryption equipment procured by the Irish Department of Communications from the Swiss company Crypto AG, used for securing high-level confidential correspondence, had a "back-door"⁸ to realise that ethics do not play a role in the likelihood of such events.

⁸In the Crypto AG case [Spi96], it was made possible for certain national security agencies to intercept Irish

Such attacks may occur with or without the knowledge of the vendor, and may take place at equipment design time, fabrication time or programming time. At every stage in the process there is abundant opportunity for the well-resourced to insert an undetectable modification capable of subtly controlling the result of an Irish election. The only effective safeguard against this is an independent audit, *i. e.* a voter-verified record.

2.2.9 The voting machine seals are vulnerable

The voting machines have two seals which are intended to prevent access to the electronics. There is no evaluation of the effectiveness of the seals. Many commercial seals are vulnerable to some embarrassingly low-tech attacks (for instance, cutting through the glue with a sharp knife and leaving the security substrate intact). Given enough time, a seal may be broken and repaired. If the attacker has suitable resources, a counterfeit seal may even be manufactured. Finally, of course, seals may be stolen (or bought) and applied by attackers instead of by “authorised persons”.

Seals are an effective way of protecting ballot boxes for short periods with little supervision, and ballot boxes are only vulnerable to attack while they contain ballot papers. Voting machines, on the other hand, must be protected from attack all year round.

It should be remembered that the seal is no better than the inspection that preceded the sealing. If the machine might have been compromised, then sealing it will only give a false sense of security to anyone inspecting the seal. It has been established in section 2.2.4 that a full inspection is practically impossible; it must be asked what kind of inspection a returning officer will do before deciding to seal a machine.

2.3 The ballot record can be changed during counting

In this subsection, we discuss several ways that the contents of a ballot module could be changed after removal from the voting machine.

- The Nedap/Powervote system replaces ballot boxes, which are solid metal boxes occupying 27 litres of physical space and weighing several kilograms, with ballot modules small and light enough to fit in a pocket. A ballot module is manifestly more prone to loss and prestidigitation. Indeed, when representatives of the Department of the Environment were asked about the purpose of the backup ballot module (given its uselessness as a means of fault-tolerance or redundancy in the computing sense) one of the reasons given was loss in transit of the primary module.

It seems remarkable that in light of this major difference between the systems, more information and detail on the handling of the ballot modules is not available. It may be entirely practical for certain persons to replace ballot modules, or to alter their contents prior to the counting procedure. Electronic recording systems inherently feature fast, easy means for altering content. It would not be difficult to construct a device capable

communications and render them intelligible without detection.

of reading a ballot module and altering its contents just enough to alter the course of an election.

- In computing security, there is a well-known form of attack often referred to as a “man in the middle attack”. In such an attack, rather than going to the trouble of subverting and controlling an entire system one merely intercepts communications to or from the system, modifying as necessary. Such attacks are typically trivial, cheap and successful.

The Nedap/Powervote system is prone to many such attacks. For example if the programming/reading unit (PRU) is modified, replaced or reprogrammed it is possible to change the contents of the ballot module prior to its being processed by the counting software. Even easier would be to replace the counting software with identical-looking software which gives different results; such attacks can be easily rendered practically undetectable and would not require much time or resources. Many computer viruses (especially macro viruses) work this way when doing their damage, and they could easily be adapted to access a Microsoft Access database instead of a Microsoft Word document.

2.4 The counting is not necessarily correct

Subsection 2.4.1 explains why counting errors that occur will go undetected under the proposed system. Subsection 2.4.2 reminds the reader how such errors may arise through programming error or malfunction, and subsection 2.4.3 explains how errors arising from determined efforts at tampering will go undetected.

2.4.1 Lack of supervision opens counting to new risks

As discussed in section 2.1.3, the counting process in our elections has always been a public affair, where tallymen and other observers could see that nothing interfered with the process to any significant extent. This essential check is removed with the adoption of all-electronic counting.

When results are announced from the counting PCs, they will generally be accepted if they are in line with expectations; that is, if they are similar to the results of informal opinion polls. This is simply because without a supervised counting process, there is no other basis on which to judge the correctness of the counting without independently examining the ballot modules.

Real election results have occasionally been quite different from the exit poll predictions. If and when an election is conducted in an atmosphere of intimidation, when exit polls are likely to be misleading, a counting error which tends the official result toward the predicted result will likely go unnoticed.

2.4.2 Errors in counting PCs are likely

As discussed in section 2.2.1 above (*q. v.*), software is inherently vulnerable to programming error, and Section 2.2.2 explains that computing hardware is vulnerable to spontaneous bit in-

version (known informally as “cosmic rays”). Those sections also set out some reasons why the Powervote counting system is particularly vulnerable to both types of error.

2.4.3 Tampering with counting PCs is possible

Section 2.3 discussed the vulnerability of the counting PCs to “man in the middle” attacks which allow modification of vote records or results without the knowledge of the people using or supervising the machines.

The only practical obstacle to carrying out such an attack is that the malicious program must be introduced into the count PCs by the same conduit used by the “official” software — that is, an attacker must compromise the official software install disc or one of the computers used to develop or distribute the official software.

The claim that the count PCs are not vulnerable to these attacks is a mere fig-leaf: computer viruses circulated among equally “security hardened” PCs in the 1980s without need of any networks or modems. Viruses, as the name implies, are parasitically attached to host programs, and are copied and run wherever the host programs are copied and run. Unless an election hacker is unwise enough to release the virus widely, it will not be added to the list of viruses detected by antivirus utilities. This is an important point; antivirus software **will not** detect any virus that is not already known to the antivirus companies. A specially-tailored virus could easily accompany the Powervote software from its development PCs to the counting PCs without anyone detecting its presence.

2.5 Inadequate consultation

It is a truism among computer security experts that “security through obscurity is no security,” which means that attempting to keep the details of an insecure system secret leaves the system insecure. Peer review is as essential in computer security schemes as it is in scientific research. When peer review is neglected or avoided, the system designers are taking the risk that a major flaw may be discovered by a third party before the designers become aware of it but after the system is widely deployed.⁹

The proposed Nedap/Powervote e-voting system has not undergone peer review. Instead, company confidentiality has been advanced as a reason for keeping the details secret. In these circumstances, it is possible that one or more critical flaws exist in the secret design which will not come to light until someone acquires the design illegally. The design can be obtained by examining the system in detail, a process called *reverse engineering*.

Almost invariably in the field of electronic voting, increased peer review and consultation have led to recommendations for a VVAT. Most recently, a report [coLGT04] by the Local Government and Transport Committee of the Scottish Parliament noted that

⁹For instance, this happened to the A5 algorithm which supposedly secured the confidentiality of GSM communications.

the technology relating to e-voting is less advanced (than e-counting), and so the introduction of e-voting is likely only to be a longer term possibility. There appear to be a number of challenges to overcome before e-voting could be introduced, not least that of ensuring confidence in a system which will dispense with actual ballot papers.

and recommended the introduction of electronic counting only, a system which will maintain a voter-verified audit trail.

In Ireland, the lack of adequate consultation and peer-review in the process is compounded by the astonishing lack of openness on the part of the Department of the Environment, Heritage and Local Government to facilitate the provision of information and to deal with issues raised. That researchers have had to repeatedly use Freedom of Information legislation and expend enormous amounts of time and money obtaining information on what should be the most public and accountable of processes is indicative of an attitude and mind-set which does not lead to well-rounded well-specified design requirements.

3 Secrecy

Although there are existing secrecy problems in Irish elections and referenda — not least the vulnerabilities to secrecy presented by postal voting — electronic voting does bring with it new threats and potentially allows for an unprecedented level of compromise to electoral secrecy.

In the words of Chief Justice Ó Dálaigh, speaking for the Supreme Court in *McMahon v. Attorney General* [1972] IR 69:

Limited secrecy is not secrecy: it is something less than secrecy. . . . Article 16.1.4° speaks of voting by secret ballot. The fundamental question is secret to whom? In my opinion there can be only one plain and logical answer to that question. The answer is: secret to the voter. It is the voter's secret. It is an unshared secret. It ceases to be secret if it is disclosed. The Constitution guarantees the voter that his vote will be secret. In my opinion the Constitution requires that nothing shall be done which would make it possible to violate that secrecy. . . . To my mind the conclusion is inevitable that any contrivance or method by which the ballot can be identified and the voter exposed is unauthorised and no legislative enactment can give it the force of law. . . . In my opinion a voting system which permits a state official to note the number of the ballot paper of every voter in the State, and which requires the information to be stored for a full year after the poll, of itself offends against the spirit and substance of the declaration that voting shall be by secret ballot. Under such a system the fear of disclosure which secrecy is designed to drive away is ostentatiously retained. Constitutional rights are declared not alone because of bitter memories of the past but no less because of the improbable, but not to be overlooked, perils of the future.

In this section we show how the design of the voting machine software violates secrecy by permitting the recovery of a partial order of recorded votes to match with a partial order of observed voters; we show how the possibly protected right to abstain is violated (if indeed it is a right) by the system; and we show how the system's lack of transparency may be used to intimidate voters by suggesting that the voting machines may be facilitating the task of matching votes to voters.

3.1 Votes are not stored randomly as required

The machine records ballots in a pseudorandom¹⁰ order in the ballot module. The intention is that it should be impossible to tell from an examination of the ballot module information about the order in which the ballots were cast. This is important, as it would threaten secrecy if a list of voters seen using a machine could be correlated with a list of votes recovered from that machine's ballot module.

¹⁰It's impossible to get truly random numbers from a deterministic computer; pseudorandom number sequences are used which have statistical characteristics similar to those of real random number sequences.

However, the pseudorandom storage strategy used in the Nedap/Powervote voting machines does not completely obscure the information about storage. Each ballot can be stored in one of only two places, and generally ballot records stored in the middle of the block were cast before ballot records stored at the ends of the block.

If knowledge of the voting preferences of several voters is available, then it may be possible to reconstruct the order in which ballots were cast with a high degree of confidence. In particular, if the preferences of the first and third voters are known and unique, then the preferences of the second voter can be reliably discovered regardless of how many votes are recorded on the ballot module.¹¹

It should be kept in mind that in a voting system utilising PRSTV it is not hard to choose one's vote such that it be extremely likely to be unique and hence readily identifiable. Thus it is entirely feasible for determined persons to be sure which vote was cast first, third and so on merely by ensuring they are the first, third person to vote and so on.

3.2 Voters cannot abstain in secret

It may be argued that voters have a right to abstain in public elections just as committee members have a right to abstain in votes on motions, and for the same reasons.¹² If so, then it can hardly be denied that voters have a right to abstain *in secret* — that is, to fail to cast a ballot while giving the outward impression of casting one. This is of particular importance in a system where the names of persons turning up to poll are readily available.

The proposed electronic voting system lacks a way to facilitate such voters. It is impossible to abstain without making officials aware of this fact.¹³ In a referendum, for instance, abstainers are compelled to cast a vote affecting the outcome, against their conscience, or else risk disclosure of their choice.

3.3 Voting machines can be programmed to violate secrecy

Malicious programming or voting machine design could affect secrecy, by storing the ballots in such a way that the order in which they were cast can be deduced from the ballot module. The ballots in order can then be matched against a list of voters seen using the machine in question. Subsection 3.1 discusses the problems with the present storage strategy; however even if those problems did not exist, there is no way for a voter to verify that the machine is correctly programmed to properly randomise the storage. A voter may reasonably be convinced that the order of votes can be recovered from a statistical analysis of the ballot module, and it

¹¹The PTB report explains the storage strategy. The first ballot record is stored in a pseudorandom position. The second record is stored in a neighbouring location (either one up or one down, chosen pseudorandomly), and the third is stored in turn adjacent to one of the first two records. For instance, if the first ballot is found at position 50 and the third is found at position 51, then the second ballot must be in position 49.

¹²Abstaining should be distinguished from intentional spoiling, though the mechanism is often the same.

¹³Upon a 'null vote' being registered an official must reset the voting machine.

may well be a true fear rather than an irrational one. It was established in *McMahon v. Attorney General* that fear of disclosure of one's vote must not be "ostentatiously retained".

4 Testing

In considering the testing of the chosen Nedap/Powervote system it is important to note that in many cases, while we consider the testing inadequate and not fully relevant, this is primarily a reflection on the specification for the testing performed and need not imply any criticism of the competency of the various testing agencies.

4.1 Review of consultants' reports

The Department commissioned a number of reports relating to *parts* of the Nedap/Powervote system;

- The reports by KEMA Quality BV and TNO were not concerned with either the accuracy or the security of the machines.
- The reports by PTB examined many aspects of the voting machine. In particular, PTB were asked to evaluate the system with respect to security criteria provided by the Department. Those criteria inadequately grouped all attackers together in the phrase “unauthorised persons” which was not defined. PTB interpreted this phrase as excluding all persons able to seal the voting machine (see section 2.2.9). PTB volunteered in [KGSS03, page 11] that “An exchange of the ROM chips including fraudulent presentation of the correct checksums cannot be avoided by software but by means of sealing only” which indicates clearly that tampering by an “authorised person” under cover of official maintenance will pass a returning officer’s inspection.
- Electoral Reform Services (ERS) tested the count software using their database of PRSTV ballots and compared the results with their reference implementation of PRSTV. Discrepancies between the two results indicate a fault in either or possibly both implementations. Some versions of the submitted software *failed these tests* [WW03, page 5] but they did eventually receive a version of the software which worked with all of their test data, namely IES v121. In their report, ERS clearly state the limitations of this comparison test and also list aspects of the count software that were not tested. They conclude that it is “difficult to precisely quantify” the risk of an incorrect result but they estimate the chance of an error to be between 1 in 1,000 and 1 in 10,000. This estimate is not supported by any statistical or risk analysis or any other scientific basis.
- The Nathean report showed the result of reviewing 70,000 lines of source code specially written for the Irish electoral system; a further 130,000 lines which are not specific to Ireland did not receive the same level of attention. No mention was made of inspection of machine code.

In some cases [Tec03], Nathean raised “No Issues” with sections of code they themselves described as “Still mostly Dutch making it difficult to understand the processes”. Although Nathean later procured the services of a Dutch-speaking code reviewer, some of the attitudes evident in various Nathean reports are not indicative of a professional and thorough code audit.

- The Zerflow report examined security procedures in the polling station alone. In a report [Eng02] in March 2002, Zerflow reference some issues raised at a meeting with Peter Greene (Department of the Environment Franchise section), including:

What actions and processes are audited, how is the audit trail linked, and who has access to that audit?

In the case of a dispute, is there any sort of a recount, or manual input via audit trail?

How does the system confirm that each vote has been accepted (or rejected) and recorded?

How does the local poll centre staff know, and verify that the system is working correctly?

What is the fault tolerance, and how has it been validated?

In the case of a dispute . . . does that election stand, is there a method of checking, can the audit trail provide any further information?

These questions have **never** been answered to our satisfaction, and we submit that the only means to answer them is to provide a voter-verified audit trail.

None of the reports reached a conclusion on the overall security of the system (as distinct from a component of the system). No report mentioned voter verification. No reference was made to any computer science research into electronic voting criteria or security. The security assessments undertaken are limited and inadequate, and are not a satisfactory basis for a finding of accuracy of the system.

4.2 Lack of adequate end-to-end testing

In any computer system (indeed, in any complex system), integration testing often indicates problems where tests of individual components did not find problems. This is because the interaction between the components is complex and often unanticipated by the developers and testers of each component. Frequently, misunderstandings and ambiguities in design documents lead to subtle incompatibilities between components which are supposed to work well together.¹⁴ System integration is a complex and expert task which cannot be done well if rushed.

The proposed electronic voting system has undergone *no* significant integration test. The most extensive test reported to date is the feeding of the Buncrana UDC votes (2,483 ballots) into a single machine, the votes on the machine's module were then counted using the IES count software, which was then observed to give credible results. This is not a substitute for a real end-to-end test scheme, which should adequately mimic real-world uses of the system.

The use of electronic voting at previous elections and a referendum is often cited as evidence of prior end-to-end testing, however it should be clear that the Nedap/Powervote system used

¹⁴A famous example of this is the NASA Mars Climate Orbiter, whose loss over Mars was eventually attributed to confusion by two different groups over whether thrust values were in newtons or pounds.

previously is not the same as the system chosen for use in June. In particular the facilitation of multiple polls, larger displays and different count software mean there are substantial differences. There is also no independent voter-verified record of voters intentions from the prior elections and referendum, and so it impossible to know that votes were recorded correctly.

4.3 Inadequate security criteria

The security criteria for any complex system should be drawn up by an expert; security decisions made by amateurs often fail because of fuzzy thinking and a less-than-formal approach. Security criteria for electronic voting have been drawn up and published by Bruce Schneier [Sch00], Peter Neumann [Neu93] and Rebecca Mercuri [Mer01], among others. Nonetheless, the Department appears to have drawn up their own security criteria without expert assistance.

The criteria provided to the PTB for testing included several security criteria, but they were all of the form “Any alteration of X by an unauthorised person should be detected.” The definition of “unauthorised” is difficult to settle in any system, despite the layman’s instinct that it’s obvious; nevertheless, no attempt is made to elaborate on what it means here. PTB decided to interpret it as excluding any person capable of sealing the machine or causing a machine to be sealed. From the voter’s perspective, no person is authorised to rig a voting machine; it is the nature of the alteration which should be tested, rather than the person making the alteration (the machine would not have passed such a criterion without some form of voter verification).

A professional security evaluation would have enumerated the classes of person with an interest in attacking the system, estimated the resources and methods available to each class, anticipated the potential attacks from each class, and evaluated the system’s ability to prevent, detect, and recover from each attack. From this evaluation of the threat model, the risks could be estimated. Although individual security experts may deviate to varying degrees from this methodology, it illustrates the rigour which is typical of a serious security evaluation, and which is notably absent from the Department’s evaluation process.

It is unsurprising that the resulting system has inadequate security, and that therefore the accuracy of its results is in doubt.

4.4 Unreviewed machine code

The Government’s evaluation of the proposed system has included examination of the source code¹⁵ but not the machine code. In [Tho84], Ken Thompson described an attack he made on a system of his own design in which the incriminating code did not appear in the source code, yet it persisted in the machine code. This means that computer systems can be made to behave in ways undetectable by source code review.

It is known that the delivered machine code for the counting PCs contains code which does not

¹⁵Software engineers do not directly encode the instructions for a computer; rather, they write precise but readable descriptions called *source code* which are then mechanically translated into the *machine code* that the computer’s processor can execute directly.

correspond to any part of the source code; rather, it is inserted by a “commercial off-the-shelf” software development environment. While this is normal practice in the industry (and not a problem in audited systems), it is an ideal cover for any maliciously implanted code of the type described by Thompson. It is possible that the machine code delivered for use in the voting machines and count PCs may not correspond to the source code which was reviewed.

Any such malicious code directly threatens the accuracy and the secrecy of the system by permitting the deployment of voting machines and counting PCs which do not have the programmed behaviour expected by the reviewers of the source code. In electronic voting systems which do not include a VVAT, it is necessary to review the entire machine code in order to detect any such code. Machine code review is a task so difficult, expensive and prone to error, that it is practically impossible.

5 Conclusion

The task of the Commission is similar to the task of the Supreme Court in an invocation of the Article 26 procedure — the system is to be evaluated in a vacuum of facts and on a limited timescale, and yet the consequences of an inappropriate approval would have serious repercussions.

In [W⁺96, p. 545], James Hamilton wrote:

Secondly, it is difficult to avoid the conclusion that the Supreme Court, notwithstanding that Bills enjoy the presumption of constitutionality in Article 26 cases, is in practice very ready to strike Bills down. This is not only because the court is aware that once approved the law can never be challenged again, but also because the common law tradition is not comfortable with an abstract examination of a Bill in the absence of a plaintiff, facts and a real allegation that a wrong has been suffered.

If the court can envisage any circumstances which might exist where injustice could arise from a measure in a Bill, they are likely to strike it down, even though the likelihood of these circumstances arising might be fairly remote.

If and when the electronic voting system is defrauded, the evidence of the fraud will likely be concealed by the lack of transparency in the system. If and when it fails due to innocent error, only the most catastrophic errors, such as negative vote totals, would be detected. Errors or fraud which were detected would be uncorrectable, because no evidence of the original votes would remain. Although there will be no statutory bar to re-examining the system in the High Court when a controversy arises in the course of an election, there will be no evidence which might distinguish a vexatious action from a valid one. A serious injustice may arise as a result.

We therefore respectfully submit that if the Commission decides that circumstances might exist whereby election results may be inaccurate, then the system must be rejected, even though the likelihood of those circumstances may appear remote.

We also submit that the threats to the security of the system are both realistic and practical, and that there is therefore a substantial risk that they will develop into attacks on the integrity of an election. It is impossible to say how much time will elapse before the first such attack occurs; it will depend on the consciences of those who will find themselves in a position to make an attempt. If history is a useful guide, it is better not to be too optimistic on this point.

For any potential attacker weighing up the consequences of detection versus the potential benefit of successful fraud, unaudited electronic voting offers significantly lower risks and higher pay-offs than any system that includes voter-verification.

In summary, the unaudited electronic voting system is subject to a broad range of practical threats and we submit that it is unsafe for use until such time as an independent voter-verified audit trail is added. A voter-verified audit trail is the **only** means by which the accuracy of an electronic voting system may be assured.

References

- [ACM] ACM U.S. Public Policy Committee. E-voting technology and standards. <http://www.acm.org/usacm/Issues/EVoting.htm>.
- [Bro75] Frederick P. Brooks, Jr. *The Mythical Man-Month — Essays on Software Engineering*. Addison-Wesley Publishing Company, 1975.
- [Cho03] David Cho. Fairfax judge orders logs of voting machines inspected. *The Washington Post*, page B01, Nov 2003. 2003-11-05.
- [coLGT04] The Scottish Parliamentary committee on Local Government and Transport. Stage 1 report on the Local Governance (Scotland) Bill — 2nd report 2004 (session 2), Mar 2004.
- [Eng02] Colin English. Electronic voting security assessment for Department of Environment. Technical report, Zerflow Information Security, Mar 2002. <http://evoting.cs.may.ie/Documents/ZerflowReport.pdf>.
- [Fis96] Charles Fishman. They write the right stuff. *FastCompany*, 06, Dec 1996. <http://www.fastcompany.com/online/06/writestuff.html>.
- [KGSS03] G. Kilz, N. Greif, H. Schrepf, and D. Saborrosch. Test report 2 — voting machine ESI2 — software for elections in Ireland. Technical Report PTB-8.302-PB-04.03-V15, Physikalisch-Technische Bundesanstalt, Sep 2003. <http://www.electronicvoting.ie/pdf/PTB%20test%20rpt%202%20-%20sept03.pdf>.
- [KP99] Brian W. Kernighan and Rob Pike. *The practice of programming*. Addison-Wesley professional computing series. Addison-Wesley, Feb 1999.
- [Mal98] David Malone. Irish proportional representation election counting code, 1998.
- [Mer92] Rebecca T. Mercuri. Physical verifiability of computer systems. In *5th International Computer Virus and Security Conference*, March 1992.
- [Mer01] Rebecca Mercuri. Statement on electronic voting. <http://www.notablessoftware.com/RMstatement.html>, 2001.
- [Neu93] Peter G. Neumann. Security criteria for electronic voting. In *Proc. 16th National Computer Security Conference*, Baltimore, Maryland, Sep 1993. NIST/NCSC. <http://www.csl.sri.com/users/neumann/ncs93.html>.
- [Pow03] Powervote. Powervote/nedap website. http://www.election.nl/bizx_html/IVS-GB/, 2003.
- [Sch00] Bruce Schneier. Voting and technology. *Crypto-Gram*, 00(12), Dec 2000. <http://www.schneier.com/crypto-gram-0012.html#1>.
- [Soc04] Irish Computer Society. The ICS calls for audit trail in e-voting system. <http://www.ics.ie/article-027.shtml>, Mar 2004.

- [Spi96] Wer ist der befugte vierte? — geheimdienste unterwandern den schutz von verschlüsselungsgeräten, 1996. English translation at http://taint.org/2002/12/11/150544a_mail.html.
- [Tec03] Nathean Technologies. Code review of ies build 0111. <http://www.electronicvoting.ie/pdf/Nathean%20Code%20Review%20Dec03.pdf>, 2003.
- [Tho84] Ken Thompson. Reflections on trusting trust. *Communications of the ACM*, 27(8):761–763, Aug 1984. A. M. Turing Award lecture; <http://www.acm.org/classics/sep95/>.
- [W⁺96] Dr. T. K. Whitaker et al. *Report of the Constitution Review Group*. The Stationery Office, May 1996.
- [WW03] J. Wadsworth and B. Wichmann. Report on Irish STV software testing. Technical report, Electoral Reform Services, Dec 2003. <http://www.electronicvoting.ie/pdf/ERS%20software%20validation%20report%202003.doc>.

Appendix A

Contributors

↔ Margaret McGaley, BSc

Margaret completed her BSc in Computer Science and Software Engineering at NUI Maynooth in 2003. Her undergraduate thesis was a study of electronic voting, with particular emphasis on the system being introduced in Ireland. Margaret is currently working towards her PhD, also on electronic voting, at NUI Maynooth. She is funded by IRCSET under the EMBARK initiative.

↔ Colm MacCárthaigh

Colm is a professional software developer, systems administrator and network engineer with HEAnet, the National Education and Research Network. Occasionally Colm contributes to Apache, the webserver used by over 70% of websites on the internet, presents papers on System Administration, researches computing security and is a member of the System Administrators' Guild of Ireland. Colm has found security and coding flaws in many well-deployed systems including the Solaris operating system produced by Sun Microsystems.

↔ Adrian Colley, BA(Mod), MACM

Adrian graduated from Trinity College, Dublin with a B.A. in Computer Science in 1994. Since then he has worked as a software engineer (with emphasis on computer security and software development) in Ireland and the US. He is a member of the Association for Computing Machinery and a founding member of the System Administrators' Guild of Ireland.

↔ Dr John Pelan, BSc, MSc, PhD, MInstP, CPhys, MBCS

John holds a BSc in Applied Physics from Dublin City University together with an MSc in Computational Science and a PhD in Computational Atomic Physics both from The Queen's University of Belfast. He is a member of the Institute of Physics, the British Computer Society and the ACM. John is currently a senior systems administrator at University College London.

↔ Catherine Ansbro, BMA, MMA, DMA (ABD)

Catherine is a graduate of Indiana and Johns Hopkins universities, USA, a participant in the Northwest Enterprise Platform Program (Sligo IT and Letterkenny IT) as well as a director of Space Exploration Ltd., RealView Innovations Ltd. and Ethereal Technologies Inc., Kingsland, Boyle, Co. Roscommon.