Liability and Computer Security: Nine Principles

Ross J Anderson
Cambridge University Computer Laboratory
Email: rja14@cl.cam.ac.uk

Abstract. The conventional wisdom is that security priorities should be set by risk analysis. However, reality is subtly different: many computer security systems are at least as much about shedding liability as about minimising risk. Banks use computer security mechanisms to transfer liability to their customers; companies use them to transfer liability to their insurers, or (via the public prosecutor) to the taxpayer; and they are also used to shift the blame to other departments ("we did everything that GCHQ/the internal auditors told us to"). We derive nine principles which might help designers avoid the most common pitfalls.

Introduction

In the conventional model of technology, there is a smooth progression from research through development and engineering to a product. After this is fielded, the experience gained from its use provides feedback to the research team, and helps drive the next generation of products:

RESEARCH → DEVELOPMENT → ENGINEERING → PRODUCT

This cycle is well known, and typically takes about ten years. However, the product's failure modes may not be immediately apparent, and may even be deliberately concealed; in this case it may be several more years before litigation comes into the cycle. This is what happened with the asbestos and tobacco industries; many other examples could be given.

RESEARCH → DEVELOPMENT → ENGINEERING → PRODUCT → LITIGATION

Now many computer security systems and products are designed to achieve some particular legal result. Digital signatures, for example, are often recommended on the grounds that they are the only way in which an electronic document can be made acceptable to the courts. It may therefore be of interest that some of the first court cases involving cryptographic evidence have recently been decided, and in this paper we try to distil some of the practical wisdom which can be gleaned from them.

Civilian Uses of Cryptography

Cryptography was originally a preserve of governments; military and diplomatic organisations used it to keep messages secret. Recently, however, cryptographic mechanisms have been incorporated in a wide range of commercial systems. Automatic teller machines (ATMs) were the pioneers, and much of commercial cryptology was developed in the late 1970's and early 1980's in order to tackle the real or perceived security problems of ATM systems [MM].

This technology has since been applied to many other systems, such as lottery terminals, prepayment electricity meters, satellite and cable TV decoders, burglar alarms, membership cards, access control devices and road toll tokens. Most of these devices use cryptography to make the substitution of bogus tokens more difficult, and thus protect revenue or assets; with millions of them being sold every year, it was inevitable that the courts would sooner or later have to assess the evidence they can provide, and this is now starting to happen.

Since early 1992, we have advised in a number of cases involving disputed withdrawals from ATMs. These now include five criminal and three civil cases in Britain, two civil cases in Norway, and one civil and one criminal case in the USA. Since ATMs have been in use the longest, and are an obvious target of crime, it is not surprising that the first real legal tests of cryptographic evidence should have arisen in this way.

All our cases had a common theme of reliance by one side on claims about cryptography and computer security; in many cases the bank involved said that since its PINs were generated and verified in secure cryptographic hardware, they could not be known to any member of its staff and thus any disputed withdrawals must be the customer's fault.

However, these cases have shown that such sweeping claims do not work, and in the process have undermined some of the assumptions made by commercial computer security designers for the past fifteen years.

At the engineering level, they provided the first detailed threat model for commercial computer security systems; they showed that almost all frauds are due to blunders in application design, implementation and operation [A1]: the main threat is not the cleverness of the attacker, but the stupidity of the system builder. At the technical level, we should be much more concerned with robustness [A2], and we have shown how robustness properties can be successfully incorporated into fielded systems in [A3].

However, there is another lesson to be learned from the "phantom withdrawal" cases, which will be our concern here. This is that many security systems are really about liability rather than risk; and failure to understand this has led to many computer security systems turning out to be useless.

Using Cryptography in Evidence

We will first look at evidence; here it is well established that a defendant has the right to examine every link in the chain of evidence against him.

- One of the first cases was R v Hendy at Plymouth Crown Court. One of Norma Hendy's colleagues had a phantom withdrawal from her bank account, and as the staff at this company used to take turns going to the cash machine for each other, the victim's PIN was well known. Of the many suspects, Norma was arrested and charged for no good reason other than that the victim's purse had been in her car all day (even although this fact was widely known and the car was unlocked).
  She denied the charge vigorously; and the bank said in its evidence that the alleged withdrawal could not possibly have been made except with the card and PIN issued to the victim. This was untrue, as both theft by bank staff using extra cards, and card forgery by outsiders were known to affect this bank's customers. We therefore demanded disclosure of the bank's security manuals, audit reports and so on; the bank refused, and Norma was acquitted.
- Almost exactly the same happened in the case R v De Mott at Great Yarmouth. Philip De Mott was a taxi driver, who was accused of stealing £50 from a colleague after she had had a phantom withdrawal. His employers did not believe that he could be guilty, and applied for his bail terms to allow him to keep working for them. Again, the bank claimed that its systems were secure; again, when the evidence was demanded, it backed down and the case collapsed.

Now even the banks admit an error rate of 1 in 34,000 for ATM systems [M], and it follows that a country like Britain with 10^9 ATM transactions a year will have 30,000 phantom withdrawals and other miscellaneous malfunctions; if 10,000 of these are noticed by the victims, and the banks deny liability, then perhaps a few hundred cases will be referred to the police. Even though the police often 'file and forget' difficult cases (especially where small sums are involved and there has been no physical injury to anyone), it is not surprising that we have seen a handful of dubious prosecutions each year.

Thankfully, there now exists a solid defence. This is to demand that the Crown Prosecution Service provide a full set of the bank's security and quality documentation, including security policies and standards, crypto key management procedures and logs, audit and insurance inspectors' reports, test and bug reports, ATM balancing records and logs, and details of all customer complaints in the last seven years. The UK courts have so far upheld the rights of both criminal defendants [RS] and civil plaintiffs [MB] to this material, despite outraged protest from the banks. It is our experience that when this disclosure is granted in time, then it is highly likely that the bank will withdraw its cooperation and the case will collapse.

Of course, this defence works whether or not the defendant is actually guilty, and the organised crime squad at Scotland Yard has expressed concern that the inability of banks to support computer records could seriously hinder police operations.

In a recent trial in Bristol, for example, two men who were accused of conspiring to defraud a bank by card forgery threatened to call a banking industry expert to say that the crimes they had planned could not possibly have succeeded [RLN]. If this had been believed, then they might well have been acquitted; it is not an offence to conspire to do something which is physically impossible.

However, a journalist from a Sunday newspaper helped us to destroy this ingenious defence; after we had told her the principle of the proposed attack (but not any details), she managed to successfully alter an ATM card issued by that bank, and thus discredit the defence expert [L]. Indeed, the information we gave her was available in a document which circulated widely in the UK prison system [S] and of which the defence expert should have been aware.

Thus the first (and probably most important) lesson is this:

> **Principle 1:** Security systems which are to provide evidence must be designed and certified on the assumption that they will be examined in detail by a hostile expert.

This should have been obvious to anybody who stopped to think about the matter, yet for many years nobody in the industry (including the author) did so. Thanks to the difficulty of getting legal aid to cover expert witnesses' fees, it is only recently that such cases have started to be well fought.

These contests could have wider implications, as many banking sector crypto suppliers also sell equipment to governments. Have their military clients stopped to assess the damage which could be done if a mafioso's lawyers, embroiled in a dispute over a banking transaction, raid the design lab at six in the morning and, armed with a court order, take away all the schematics and source code they can find? Pleading national security does not work - in a recent case, lawyers staged just such a dawn raid against Britain's biggest defence electronics firm, in order to find out how many PCs were running unlicensed software.

Using the Right Threat Model

Another problem is that many designers fail to realise that most security failures occur as a result application and management blunders, and rather than concentrating on producing a well engineered system they may instead pin their faith on some particular 'silver bullet'. This might be some new cryptographic algorithm or protocol, a delivery mechanism such as a smartcard, or a set of standards such as ITSEC.

This is illustrated by a current ATM dispute in Norway. Norwegian banks spent millions on issuing all their customers with smartcards, and are now as certain as British banks (at least in public) that no debit can appear on a customer's account without the actual card and PIN issued to the customer being used. Yet a number of phantom withdrawals around the University of Trondheim have undermined their position.

In these cases, cards were stolen from offices on campus and used in ATMs and shops in the town; among the victims are highly credible witnesses who are quite certain that their PINs could not have been compromised. The banks refused to pay up, and have been backed up by the central bank and the local banking ombudsman; yet the disputed transactions (about which the bank was so certain) violated the card cycle limits. Although only NOK 5000 should have been available from ATMs and NOK 6000 from eftpos, the thief managed somehow to withdraw NOK 18000 (the extra NOK 7000 was refunded without any explanation) [BN].

This problem with the card cycle limit makes it clear that there was a problem with the application software. Now the victim cannot reasonably be expected to establish whether this problem lies in the card, in the reader, in the network, in the settlement system, or in the bank branch; it might even lie in the manual procedures for card and PIN issue or in some subtle combination. What is known is that blunders are common in the design of large systems, and many examples of unexpected application programming and management failures were noted in our technical survey of ATM crime [A1] [A2].

This survey showed that the real threat model for payment systems is that blunders get exploited in an opportunistic way. Although military intelligence agencies may have the experts and the money to carry out technical attacks on algorithms and operating systems, most crime is basically opportunist, and most criminals are both unskilled and undercapitalised; thus most of their opportunities come from the victim's mistakes. This threat model has since been further confirmed by a study of attacks on prepayment electricity meter systems [AS]; here too, security failures resulted from blunders in design and management, which some subscribers found ways to exploit.

Principle 2: Expect the real problems to come from blunders in the application design and in the way the system is operated.

Security Goals

It may seem by now that disputed transaction cases will be lost by whichever party has to bear the burden of proof. Where the customer says, "I didn't make that withdrawal", and the bank says "You did so", then what is the court to do? If the victim is supposed to find exactly where the fault lies in the bank's system, then it is very unlikely that she will succeed. If, on the other hand, the bank is asked to establish the security of its systems, then how can this be done in the face of hostile experts?

Here it is instructive to compare the practice in Britain with that in the United States. British banks claim that their systems are infallible, in that it is not possible for an ATM debit to appear on someone's account unless the card and PIN issued to him had been used in that ATM. People who complain are therefore routinely told that they must be lying, or mistaken, or the victim of fraud by a friend or relative (in which case they must be negligent). There has recently been a cosmetic change, with the introduction of a new code of banking practice; in this, the banks say that the onus is now on them. However, when confronted with a phantom withdrawal, they consider this onus to be discharged by a statement that their computer systems were working and that no known frauds were taking place at the relevant time and place.

The US is totally different; there, in the landmark court case Judd v Citibank [JC], Dorothy Judd claimed that she had not made a number of ATM withdrawals which Citibank had debited to her account; Citibank claimed that she must have done. The judge ruled that Citibank was wrong in law to claim that its systems were infallible, as this placed 'an unmeetable burden of proof' on the plaintiff. Since then, if a US bank customer disputes an electronic debit, the bank must refund the money within 30 days, unless it can prove that the claim is an attempted fraud.

British bankers claim that such a policy would be utterly disastrous; if they paid up whenever a customer complained, there would be an avalanche of fraudulent claims of fraud. But US bankers are more relaxed; their practical experience is that the annual loss due to customer misrepresentation is only about $15,000 per bank [W1], and this will not justify any serious computer security programme. In areas like New York and Los Angeles where risks are higher, banks use ATM cameras to resolve disputes.

Another unexpected finding was the relationship between risk and security investment. One might expect that as US banks are liable for fraudulent transactions, they would spend more on security than British banks do; but our research showed that precisely the reverse is the case: while UK banks and building societies now use hardware security modules to manage PINs, most US banks just encrypt PINs in software.

Thus we conclude that the real function of these hardware security modules is due diligence rather than security. British bankers want to be able to point to their security modules when fighting customer claims, while US bankers, who can only get the advertised security benefit from these devices, generally do not see any point in buying them. Given that the British strategy did not work - no-one has yet been able to construct systems which bear hostile examination - it is quite unclear that these devices add any real value at all.

Now, one of the principles of good protocol engineering is that one should never use encryption without understanding what it is for (keeping a key secret, binding two values together, ...) [AN]. This generalises naturally to the following:

Principle 3: Before setting out to build a computer security system, make sure you understand what its real purpose is (especially if this differs from its advertised purpose).

Where there is a hidden purpose, designers should be aware of a possible problem with the rules of evidence. In the USA, computer records are usually only admissible if they are made in the normal course of business; so using the computer for an abnormal purpose can render its output useless [W2]. In the UK, too, a court has thrown out ATM evidence which was obtained by a nonstandard manipulation of the system [RS].

Shifting the Blame

The most common reason for a system to have a real purpose which differs substantially from its advertised purpose is, of course, when the system owner wishes to avoid the blame when things go wrong.

In the software industry, for example, it is standard practice to offer an installation service, whereby the vendor will send a technician to install the latest upgrade for a substantial fee. Most users save the money by installing the upgrades themselves - and in so doing lose much of their ability to sue the vendor if their files get corrupted. It is also standard practice that bespoke software houses get clients to sign off every tiny specification change before it is coded and implemented - and again, this is not so much for change control and security purposes, but to make it much harder for the poor client to sue.

Things become even more problematic when one of the parties to a dispute can use market power, legal intimidation or political influence to shed liability. There are many examples of this:

1. We recently helped to evaluate the security of a burglar alarm system which is used to protect bank vaults in over a dozen countries. The vendor had claimed for years that the alarm signaling was encrypted; in Europe, this is a requirement for class 4 risks (over $10m) and recommended for class 3 risks ($250,000 to $10m) [B1]. We found that the few manipulations performed to disguise the data could not in fairness be called 'encryption' - they could not be expected to withstand even an amateur attack. The vendor's response was to try and intimidate our client into suppressing the report
2. We have mentioned some of the tricks that software houses employ; and within organisations, similar strategies are commonplace. One can expect that managers will implement just enough computer security to avoid blame for any disaster; if possible, they will ask for guidance from the internal auditors, or some other staff function, in order to diffuse the liability
3. If there is no internal scapegoat, a company may hire consultants to draw up a security specification. Members of the academic security community

often complain that so many lucrative consulting contracts go to large, well-known consultancy firms, who often do not possess their technical skills; the dynamics of blame shifting may provide an insight into the relative merits of fame and competence when purchasing security consultancy services

4. If liability cannot be transferred to the state, to suppliers, to another department, or to consultants, then managers may attempt to transfer it to customers - especially if the business is a monopoly or cartel. Utilities are notorious for refusing to entertain disputes about billing system errors; and many banking disputes also fall into this category.

> **Principle 4:** Understand how liability is transferred by any system you build or rely on.

The Limitations of Legal Process

In the world of academic cryptography, it is common to assume that the law works with the precision and reliability of the theory of numbers. Conference papers often say things like “and so Alice raises X to the power Y, and presents it to the judge, who sees that it is equal to Z and sends Bob to jail”.

Would that the world were that simple! Even if we have a robust system with a well designed and thoroughly tested application, we are still not home and dry; and conversely, if we suffer as a result of an insecure application built by someone else, we cannot rely on beating them in court.

Lawyers are well aware that the use of technical evidence, and in particular of computer evidence, is fraught with difficulty. Most judges have a background in the humanities rather than the sciences, and may be more than normally technophobic; even where these feelings are dutifully suppressed, experienced and otherwise intelligent men can find it impossible to understand simple evidence. The author has observed this phenomenon at a number of computer trials from 1986 down to the present, and has often felt that no-one else in court had any idea what was going on. Specialist computer lawyers confirm that this feeling is not uncommon in their practice.

Consider the recent case of R v Munden, in which one of our local police constables came home from holiday to find his bank account empty, asked for a statement, found six withdrawals for a total of £460 which he did not recall making, and complained to his bank. It responded by having him prosecuted for attempting to obtain money by deception. It came out during the trial [RM] that the bank's system had been implemented and managed in a rather ramshackle way, which is probably not untypical of the small data processing departments which service most medium sized commercial firms.

– The bank had no security management or quality assurance function. The software development methodology was ‘code-and-fix’, and the production code was changed as often as twice a week.

– No external assessment, whether by auditors or insurance inspectors, was produced; the manager who gave technical evidence was the same man who had originally designed and written the system twenty years before, and still ran it. He claimed that bugs could not cause disputed transaction, as his system was written in assembler, and thus all bugs caused abends. He was not aware of the existence of TCSEC or ITSEC; but nonetheless claimed that as ACF2 was used to control access, it was not possible for any systems programmer to get hold of the encryption keys which were embedded in application code.
– The disputed transactions were never properly investigated; he had just looked at the mainframe logs and not found anything which seemed wrong (and even this was only done once the trial was underway, under pressure from defence lawyers). In fact, there were another 150-200 transactions under dispute with other clients, none of which had been investigated.

It was widely felt to be shocking that, even after all this came to light, the policeman was still convicted [E]; one may hope that the conviction is overturned on appeal.

The larger pattern here is that when a new technology is introduced, the first few cases may be decided the wrong way. This is especially the case with the criminal law, as most defendants in criminal trials rely on the legal aid system, and this has a number of well documented weaknesses [HBP]. Prosecutors can expect a series of successes against poorly defended suspects, followed at last by a defeat which may define the law (and in so doing upset an entire industry's ways of working).

It seems likely that the ATM disputes will follow the same pattern. One of the first ATM related prosecutions in the UK, that of Janet Bagwell [A1], led to a notorious miscarriage of justice: there, an innocent girl admitted theft on advice for her solicitor that her chances of a successful defence were slim, and it later turned out that the disputed transaction had been the bank's fault all along. More recently, in the Hendy and De Mott cases mentioned above, the defendants had access to expert advice in time, and were acquitted; in the Munden case, the author was only brought in as the defence expert while the trial was underway, and even then the bank has shown clear public signs of remorse that the prosecution was ever brought.

A number of changes in the law have been proposed, but not all of them will be for the better. One of the main motives for change is the large number of convictions for serious crimes, such as murder and terrorism, which have recently been overturned. Many of them involved doubtful forensic evidence, and legal aid restrictions prevent most defendants from challenging this effectively.

Also, in the area of financial crime, the inability of juries to deal with complex fraud cases has led to debate on whether 'special' juries, selected from professional people, should be reintroduced in the City. Thus the problems of computer evidence are part of a much wider problem: progress makes for increasing specialisation, and without specialist knowledge being available to all parties, there are many ways in which the legal system can come adrift.

All this has led to one of the main campaigners on this issue, Michael Mansfield QC, to call for a move to the French system of examining magistrates [C1]. However, this would mean a single expert being appointed by the court, and it seems likely that such experts would be like the defendants' expert in the Bristol case (or the Home Office explosives expert in the recent IRA cases) - a man with eminent qualifications, but unable to see faults in the systems which he had spent years helping to develop.

A final problem is that even when judicial practices do finally stabilise, they may not converge. In the USA, for example, the definition of a signature varies widely. Depending on the context, one may need an original autograph signature, or a fax may do, or a tape recording, or a stamp, or even a typewritten name [W2]. Even the passage of time is not guaranteed to sort things out: in some jurisdictions, contracts are signed at the bottom, while in others they must be initialled on every page as well; this is a throwback to nineteenth century disputes on whether typewritten documents were too easy to forge, and their fallout has persisted for a century, despite causing problems for international trade.

It is thus foolish to assume that digital signatures will end up being accepted equally in all countries, or even for all purposes in any one country. Our next principle is therefore:

Principle 5: The judicial treatment of new kinds of technical evidence may take years to stabilise, and may not converge to anything consistent.

This is well enough known to lawyers, but is usually ignored by the security community - perhaps because the remedy is to prefer mechanisms which are easy for a layman to understand. Security cameras are unproblematic; yet we would not look forward to being the first litigant to try and adduce a zero knowledge proof in evidence.

Legislation

Strange computer judgments have on occasion alarmed lawmakers into attempts to rectify matters by legislation. For example, in the case of R v Gold & Schifreen [RGS], two 'hackers' had played havoc with British Telecom's electronic mail service by sending messages 'from' Prince Philip 'to' people they didn't like announcing the award of honours; this greatly upset the Establishment and they were charged with forgery (of British Telecom's engineering password). They were convicted in the first instance, but eventually freed on appeal by Lord Lane, on the grounds that information (unlike material goods) cannot be stolen or forged. This was proclaimed by the press (and by the computer security industry) to be a hackers' charter, and the ensuing panic in parliament led to the Computer Misuse Act.

This act makes 'hacking' a specific criminal offence, and thus tries to transfer some of the costs of access control from system owners to the Crown Prosecution Service. Whether it actually does anything useful is open to dispute: on the one hand firms have to take considerable precautions if they want to use it against errant employees [A5] [C2]; and on the other hand it has led to surprising convictions, such as that of a software writer who used the old established technique of putting a timelock in his code to enforce payment [C3]. Similar laws have been passed in a number of jurisdictions, and similar problems have arisen.

But even if the state possessed the competence to frame good laws on computer issues, its motives are often dubious. Powerful lobby groups get legislation to transfer their costs to the public purse; and governments have often tried to rewrite the rules to make life easier for their signals intelligence people, without thinking through the consequences for other computer users.

For example, the South African government decreed in 1986 that all users of civilian cryptology had to provide copies of their algorithms and keys to the military. Bankers approached the authorities and said that this was a welcome development; managing keys for automatic teller machines was a nuisance and the military were welcome to the job; but of course, whenever a machine was short, they would be sent the bill. At this the military backed down quickly.

More recently, the NIST public key initiative [C4] proposes that the US government will certify all the public keys in use in that country. They seem to have learned from the South African experience, in that they propose a statutory legal exemption for key management agencies; but it remains to be seen how many users will trust a key management system which they will not be able to sue when things go wrong.

Given all these problems, our next principle is inevitable:

Principle 6: Computer security legislation is highly likely to suffer from the law of unexpected consequences.

Standards

Another tack taken by some governments is to try and establish a system of security standards. These are often designed to give a legal advantage to systems which use some particular technology. For example, to facilitate CREST (the Bank of England's new share dealing system), the Treasury proposes to amend English law so that the existence of a digital signature on a stock transfer order will create 'an equitable interest by way of tenancy in common in the ... securities pending registration' [HMT].

On a more general note, some people are beginning to see a TCSEC C2 evaluation as the 'gold standard' of commercial computer security. This might lead in time to a situation where someone who had not used a C2 product might be considered negligent, and someone who had used one might hope that the burden of proof had passed to someone else. However, in the Munden case, the bank did indeed use an evaluated product - ACF2 was one of the first products to gain the C2 rating - yet this evaluation was not only irrelevant to the case, but not even known to the bank.

The situation can be even worse when standards are promulgated which have flaws, or which conflict with each other. The problems with X.509 [BAN], the controversy over ISO 11166 [R], and the debate about the relative merits of RSA and DSA, are well enough known to; it may be that some of the things said by eminent people about DSA in the heat of the debate in 1992 [B2] will be exhumed in years to come and brandished by a jubilant defence lawyer.

In any case, it is well known that standards are used as pawns in battles for market share, and a standard which appears to be safe and well established today might be subject to a fierce challenge in a few years' time - again, the RSA versus DSA debate is a useful example here.

For all these reasons, it is imprudent to expect that the standards industry will ultimately provide an effort-free solution to all the legal problems which can affect security systems. Standards are much less stable than laws should be, and are often founded on much baser and more fickle motives.

Principle 7: Don't rely on engineering standards to solve legal problems.

A related point is that although the courts often rely on industry practice when determining which of two parties has been negligent, existing computer security standards do not help much. After all, they mostly have to do with operating system level features, while the industry practices themselves tend to be expressed in application detail - precisely where the security problems arise. The legal authority flows from the industrial practice to the application, not the other way around.

It is pure hubris for the security technical community to think that court cases should be decided by considering the merits of various encryption schemes. Of course, it is always conceivable that some future dispute will involve mutual allegations of insecurity between two EDI trading partners, and that competing expert evidence will be heard on which of two authentication schemes is easier to circumvent. However, where there is a conflict of experts, the courts tend to disregard both of them and decide the case on other evidence.

This other evidence then has to be interpreted in line with normal practice, whatever that may be. Is it usual for a Dutch banker to issue a guarantee in the form of a telex, or of a letter with two signatures? Should an Indian scrap metal purchaser draw a letter of credit to be made payable against a faxed copy of an inspection certificate, or should he stipulate the production of the original document? These are the sort of questions on which real cases turn, and they are usually decided by reference to the actual practice in a given trade.

Understanding this could have saved British and Norwegian bankers a lot of security expenditure, legal fees and public embarrassment; for in traditional banking, the onus is on the bank to show that it made each debit in accordance with the customer's mandate.

> **Principle 8:** Security goals and assumptions should be based on industry practice in the application area, not on general 'computer' concepts.

Liability and Insurance

The above sections may have given the reader the impression that managing the liability aspects of computer security systems is just beyond most companies. This does not mean that the problem should be accepted as intractable, but rather that it should be passed to a specialist - the insurer.

As insurers become more aware of the computer related element in their risks, it is likely that they will acquire much more clout in setting security standards. This is already happening at the top end of the market: banks who wish to insure against computer fraud usually need to have their systems inspected by a firm approved by the insurer.

The present system could be improved [A4] - in particular the inspections, which focus on operational controls, should be broadened to include application reviews. However, this is a detail; certification is bound to spread down to smaller risks, and, under current business conditions, it could economically be introduced for risks of the order of $250,000. It is surely only a matter of time before insurance driven computer security standards affect not just businesses and wealthy individuals, but most of us [N1].

Just as my insurance policy may now specify 'a five-lever mortise deadlock', so the policy I buy in ten years' time is likely to insist that I use accounting software from an approved product list, and certify that I manage its security features in accordance with the manual, if my practice is to be covered against loss of data and various kinds of crime.

Insurance-based certification will not mean hardening systems to military levels, but rather finding one or more levels of assurance at which insurance business can be conducted profitably. The protection must be cheap enough that insurance can be sold, yet good enough to keep the level of claims under control.

Insurance-based security will bring many other benefits, such as arbitration; any dispute I have with you will be resolved between my insurer and your insurer, as with most motor insurance claims, thus saving the bills (and the follies) of lawyers. Insurance companies are also better able to deal with government meddling; they can lobby for offensive legislation to be repealed, or just decline to cover any system whose keys are kept on a government server, unless the government provides a full indemnity.

A liability based approach can also settle a number of intellectual disputes, such as the old question of trust. What is 'trust'? At present, the US DoD 'functional' definition states that a trusted component is one which, if it breaks, can compromise system security, while Needham's alternative 'organisational' definition [N2] states that a trusted component is one which my employer allows me to trust (if it breaks and the system security is compromised as a result, I do not get fired).

From the liability point of view, of course, a component which can be trusted is one such that, if it breaks and compromises my system security, I do not lose an unpredictable amount of money. In other words:

Principle 9: A trusted component or system is one which you can insure.

References

[A1] RJ Anderson, "Why Cryptosystems Fail", in *Proceedings of the 1st ACM Conference on Computer and Communications Security* (1993) pp 215 - 227

[A2] RJ Anderson, "Why Cryptosystems Fail", in *Communications of the ACM*, November 1994

[A3] RJ Anderson, "Making Smartcard Systems Robust", to appear in *Cardis 94*

[A4] RJ Anderson, "Liability, Trust and Security Standards", in *Proceedings of the 1994 Cambridge Workshop on Security Protocols* (Springer, to appear)

[A5] J Austen, "Computer Crime: ignorance or apathy?", in *The Computer Bulletin v 5 no 5* (Oct 93) pp 23 - 24

[AN] M Abadi, RM Needham, *'Prudent Engineering Practice for Cryptographic Protocols'*, DEC SRC Technical Report no 125 (1994).

[AS] RJ Anderson, S Bezuidenhout, "On the Security of Prepayment Metering Systems" (*to appear*)

[B1] KM Banks, Kluwer Security Bulletin, 4 Oct 93

[B2] DJ Bidzos, Letter to Congress, September 20 1991; published in usenet newsgroup `comp.risks 12.37`

[BAN] M Burrows, M Abadi, RM Needham, "A Logic of Authentication", in *Proceedings of the Royal Society of London A* v 426 (1989) pp 233 - 271

[BN] Behne v Den Norske Bank, Bankklagenemnda, Sak nr: 92457/93111

[C1] S Clark, "When justice lacks all conviction", in *The Sunday Times* (31 July 1994) section 4 page 7

[C2] T Corbitt, "The Computer Misuse Act", in *Computer Fraud and Security Bulletin* (Feb 94) pp 13 - 17

[C3] A Collins, "Court decides software time-locks are illegal", in *Computer Weekly* (19 August 93) p 1

[C4] S Chokhani, "Public Key Infrastructure Study (PKI)", in *Proceedings of the first ISOC Symposium on Network and Distributed System Security* (1994) p 45

[E] B Ellis, "Prosecuted for complaint over cash machine", in *The Sunday Times*, 27th March 1994, section 5 page 1

[HBP] M McConville, J Hodgson, A Pavlovic, *'Standing Accused: The Organisation and Practices of Criminal Defence Lawyers in Britain'*, OUP (1994) reviewed by David Pannick QC in The Times, 16 August 1994, p 33

[HMT] HM Treasury, *'CREST - The Legal Issues'*, March 1994

[ITSEC] *'Information Technology Security Evaluation Criteria'*, June 1991, EC document COM(90) 314

[J] RB Jack (chairman), *'Banking services: law and practice report by the Review Committee'*, HMSO, London, 1989

[JC] Dorothy Judd v Citibank, *435 NYS, 2d series, pp 210 - 212, 107 Misc.2d 526*

[L] B Lewis, "How to rob a bank the cashcard way", in *Sunday Telegraph* 25th April 1992 p 5

[M] S McConnell, "Barclays defends its cash machines", in *The Times*, 7 November 1992

[MB] McConville & others v Barclays Bank & others, Queen's Bench Division 1992 ORB no. 812

[MM] CH Meyer and SM Matyas, *'Cryptography: A New Dimension in Computer Data Security'*, John Wiley and Sons 1982.

[N1] RM Needham, "Insurance and protection of data", *preprint*

[N2] RM Needham, comment at 1993 Cambridge formal methods workshop

[R] RA Rueppel, "Criticism of ISO CD 11166 banking - key management by means of asymmetric algorithms", in *Proceedings of 3rd Symposium on State and Progress of Research in Cryptography*, Fondazione Ugo Bordoni (1993) pp 191 - 198

[RGS] R v Gold and Schifreen, Southwark Crown Court, 1986

[RLN] R v Lock and North, Bristol Crown Court, 1993

[RM] R v Munden, Mildenhall Magistrates' Court, 8-11 February 1994

[RS] R v Small, Norwich Crown Court, 1994

[S] A Stone, "ATM cards & fraud", *manuscript 1993*

[TCSEC] *'Trusted Computer System Evaluation Criteria'*, US Department of Defense, 5200.28-STD, December 1985

[W1] MA Wright, "Security Controls in ATM Systems", in *Computer Fraud and Security Bulletin*, November 1991, pp 11 - 14

[W2] B Wright, *'The Law of Electronic Commerce'*, Little, brown & Co, 1994

This article was processed using the LaTeX macro package with LLNCS style