



Ensuring That Your Effective Use of the Internet is Safe and Secure

John R T Brazier, Professional Projects Co Ltd

Introduction

These notes supported a talk given by John R T Brazier at the Innovations 97 Conference, 15 and 16 September 1997. Whilst the conference was targeted for pharmaceutical companies, this talk had general reference, and these notes have general applicability with regard to Internet security.

The talk commenced by placing Internet security in context; it then discussed the current security and risk factors associated with Internet usage; covered some of the security technologies available; went into some depth on cryptology and its issues; and completed with the effects of Governments on cryptology. The rest of the notes are in the present tense, as supplied to the conference (some points have been briefly updated as of March 1998).

Security and the Internet

For the purposes of this talk, the widest view is taken of the Internet. It is taken to be the low-cost IP-based world-wide network that was derived from ARPANET and provides capabilities such as FTP, mail, news services, search services and the World Wide Web. This talk also takes the view that Internet security is about information protection (IP), in its widest sense. Organisations attempt to protect their information to achieve the following aims:

- Confidentiality – that only the right people can access the information; that it is protected from unauthorised disclosure (in storage, use or transit).
- Integrity – that the information is accurate and complete; that it can only be updated or modified by authorised people in approved ways.
- Availability – the information is available when required.

This is a classic model of information protection: if the three above aims are achieved, then the organisation's information can be regarded as secure (to some specified level). However, there are many risks threatening such security, and the Internet is only one of them. In fact, whilst there are real (and potentially disastrous) risks with using the Internet, these are almost certainly secondary to the biggest threat facing all companies: internal employees. A recent estimate was 70 to 80% of information security problems are caused by insiders [1].

Thus whilst the Internet does provide new ways to lose control over one's information, it should still be placed in the context of the overall business and informational risks. However, to do this means that an organisation must have an overall strategy and plan to deal with security, and have in place policies which allow the management of information security and the rational evaluation of risk. Without such plans and policies, technologies and ad hoc procedures typically provide little security. A good starting point for such planning and security management is given in [2].



Internet Security and Risks

The current state of security on the Internet as a whole is poor. It was never designed for security overall, nor were its basic protocols (IP) or operating systems (usually Unix). Whilst this is being addressed for the future (with 'IP next generation' and 'hardened' versions of Unix), the current position is due to:

- There are no security standards across the net – or too many. There are a number of proposals and technologies, but as yet there is no clear unifying set of standards.
- The Internet comprises a large number of different technologies (hardware and software), making security much more difficult due to the complex infrastructure.
- The technologies are evolving extremely rapidly. Hardware and software cycles are contracting in time; new processors, operating systems, communications support software and applications are being constantly produced. The change makes security difficult because all the new items have to be incorporated in the absence of a unified standard; in addition; the systems frequently have flaws (consider the WWW browsers Netscape and MS Explorer, and their problems).
- Some of these technologies are actively hostile to security. Concepts such as 'anonymous FTP' are clearly against normal security principles; in addition, clients can be subverted by technologies such as ActiveX and Java.
- There are hostile people on the net. There are those who wish to break security because of the academic interest or kudos (hackers), and those who have financial incentives (crackers). These people are constantly probing systems and networks for gaps in their security. Naturally, the more prestigious the target, the more effort will be put in to hack it.

These factors mean that there are a number of risks associated with using the Internet. The first is the simplest: if you connect your machines or networks to the Internet, then you immediately create the opportunity for someone to attack your systems. No connection, no attack: even if you try to set up for 'outbound only' traffic, your point of presence on the Internet can still be attacked (and possibly penetrated if your security is not tight). However, such a decision is limiting and is usually not acceptable.

As the individuals in an organisation start using the Internet, in whatever way, risks begin to rise in number, type and severity:

- FTP file downloads immediately pose a problem if uncontrolled, as an individual can pull down some form of rogue program onto the internal network (ie viruses, Trojan horses, worms, logic bombs, etc).
- Mail is another route by which viruses can get in, as attachments. This method can also circumvent many anti-viral procedures (such as the removal of PC A: drives).
- Mail also has some other, more subtle problems:
 - Junk mail, which wastes time and resources. 'Spamming' the Internet has become an endemic problem.



- Currently, E-mail has limited authentication. You may not be exchanging messages with whom you think you are, which leads to many security risks.
- There are content issues. Libel by E-mail has already led to Court cases, and the often relatively informal nature of E-mail can lead to messages that may be regretted in the future. In addition, it is unclear as to the validity of E-mails in terms of agreements and contracts
- The World Wide Web brings its own extra issues:
 - Its simplicity of use encourages people to explore, with possibly unwanted side effects.
 - It can be a great time-waster. This is perhaps acceptable to some extent, but the very structure of the web makes it easy to get diverted from one's informational goals.
 - Again, there is an authentication issue with regard to web sites. The people running the site may not be the people you think they are. In addition, 'spoofing' is possible: this is where one connects to a site, and then moves on to the target site. However, a malicious site can trap the move request, and becomes an intermediary, forwarding requests for the target site and responses back to you (known as a 'man in the middle attack', and allows both eavesdropping and data modification).
 - The use of modern browsers has led to a new problem: their capability. The desire for animation and advanced features has led to the use of technologies such as ActiveX and Java to develop browser components. In effect, each time a browser runs an ActiveX or Java object, it is downloading and running a program, with all the implications for security. Throughout 1997 security holes have been exposed in either the Microsoft Explorer or the Netscape browsers because of such capabilities, forcing the companies to scramble to release fixes. Note that Java, whilst billed as being 'safe' (as it cannot access the client disks) can still make 'denial of service' attacks on the client PCs.
- Perhaps the most unexpected risk is caused by the very success of the Internet: organisations can relatively rapidly become highly dependent on it (often first on the electronic mail connectivity, and then on the data access capabilities). It must not be forgotten that a dependency is being acquired on a system that is outside the organisation's control, has erratic service levels and little inherent data security.

The organisation can also use the Internet to provide services (be they mail routing, FTP or a web site). As soon as an organisation does provide such services, new risks come into play:

- The organisation's systems now must allow incoming traffic (it cannot be set for only 'outward bound' connections). Thus the organisation of security becomes more complex as different levels of incoming access need to be managed and controlled.
- As has already been mentioned, communications systems (such as mail) can bring in unwanted pests (such as viruses). The problem is made worse if the organisation allows large quantities of incoming information, as it can become difficult to filter and control.

Organisations now tend to think of the web server if they wish to provide a publicly accessible presence. This will usually do one (or more) of three things: (1) act as a promotional marketing site;



(2) provide informative content which is useful, and may or may not be product-related; and (3) be a full commercial site, selling products and taking payments, and thus allowing transactions.

All three types of site will allow greater or lesser levels of interactivity, although the third transaction-oriented site will always have the greatest level of two-way communications. New risks start to appear with all three types of web presence:

- In the first case, if a web site is interesting (for whatever reason) it will become an object of attention for the hacker community. For example, both the CIA and the Jurassic Park site were broken into and altered, and turned into objects of mockery (ie 'Central Stupid Agency' and 'The Lost Pond: Jurassic Duck'). Such obvious alterations could clearly damage business reputation, especially if the changes were, for example, racist or pornographic.
- Another risk is not the obvious alteration, but the subtle one. For example, crackers could break into an advertising site, and change all the telephone numbers to those of their operators. This interception of calls could have serious consequences, especially for a company doing a lot of business via the Internet.
- Commercial sites clearly must be interesting to crackers, who would like to break into transaction sites and, perhaps, collect VISA card numbers. So far no case has been found where credit card numbers have been intercepted on their way to, or stolen from, a web transaction site. Thus this is still only a potential problem, although the author believes that it will happen sooner or later: consider other card-related events, such as the publication of a legitimate card number/checksum generator program in Japan [3], and the appearance of over 10,000 legal card numbers on a US web site (probably from a bank file) [4].
- One other risk arises from the global visibility of the web site. This means that all statements should comply with the laws of, potentially, every country in the world. This may be difficult or even impossible to arrange, depending on the conditions: a look at recent events in Germany highlights some of the issues. In May 1997 Felix Somm, the head of CompuServe's German operation, was arrested with being an accessory to the dissemination of pornography [5], and since then Germany has made neo-Nazi propaganda and pornography illegal on the Internet. It has also made the service providers responsible for such content, even when uploaded outside Germany [4]. Given that the suppression of free speech (however offensive) is against the Constitution in the USA, this means that ISPs like CompuServe are facing a legal requirement for censorship in one country that is illegal in another.

Technologies for Security

Before we address the technologies that are specifically available, it is essential that there is a security policy and strategy, with resultant standards and operating procedures, within which the technology is used. This simply reiterates the point made at the start of this document. This view has practical consequences: before investing in Internet-specific security items, the following things should also be carried out:

- The personnel policies should be set up, to reduce risks from staff.
- Use the current network and operating system security. Ensure that your systems are already appropriately secure. Frequently they are not, often due a lack of understanding of the systems



and networks within an organisation, caused by the very high staff turnovers that are currently being experienced.

- If special or extra security systems already exist, use them. To give an example, there is a security technology that currently exists for web browsers called SSL, which is free and actually integrated in the browsers. Only 0.5% of web sites are using it [6].
- Make sure your backup systems are in place, are actually functioning, and that restore tests are being done regularly. In the end, backups are your only lifeline if a serious event really does take place.

Having ensured that the base systems are in place, then there are a number of technologies that are invaluable for Internet security. Probably the best known is the so-called firewall, which has a very simple aim: to separate the internal organisation's systems and networks from the external Internet, and to control the information flow and access (in both directions). It thus is, in effect, a gateway, which is designed entirely for security. The technology involved in firewalls is developing extremely rapidly, and they also offer many different capabilities and services. There are two basically different types of firewall:

- Ones which are completely proprietary, with their own hardware, operating system and software. These often tend to be low-level (in terms of the OSI model) and to be very fast and secure (if limited).
- Ones that run on standard platforms, but have their own software. They will frequently have modified operating systems, such as 'hardened' versions of Unix (although not always). These tend to be slower, and work at higher levels of the OSI model.

Firewalls are also defined in terms of how they function. Currently, three different types exist, although these categories are beginning to merge. This is because manufacturers are beginning to provide firewalls with multiple capabilities, and because the software provided is improving: increasingly one simply sets up security policies on the firewall, and how they are executed becomes relatively immaterial. The three fundamental types are:

- Packet filters: these work at the network level of the OSI stack, allowing packets through or not depending on their addresses and according to rules defined by the administrator. They are usually fast, relatively cheap, and often part of the functionality of a router. However, configuration is usually difficult and it is easy to make errors (leading to security holes).
- Proxy servers: these work at the application level of the OSI model and act as relays, forwarding user requests in one direction, and responses in the other. All links should go via the proxy, and there should be no direct flow between the external and internal networks. These are often easier to set up, but tend to be slower (degrading user response) and can be inflexible in that a proxy needs to be produced for a new application (which can lead to delays).
- Stateful Multi-Layer Inspection systems extend the concept of packet filters in that they attempt to control the information flow by analysing the communications flow at all levels of the OSI stack. With complex rule engines they attempt to make filtering decisions based on a much better 'awareness' of the applications and their communications. In effect, they try to give the speed of packet filters with the security of the proxy server, without using proxies.



Firewalls usually provide a number of services to improve security and generally support the business systems. For example, most will conceal all the internal structure of the network from the Internet by providing only one visible address, making attacks much more difficult. Many now incorporate good reporting, audit and analysis features, and some support VPNs over the Internet, where isolated networks can be logically linked, with the links encrypted for security.

In addition to firewalls, there are other tools available to help protect the internal systems. These include:

- Virus scanners. Classically, these are run on demand on individual PCs, for example to check a floppy disk entered into a PC's A: drive. In this case, they are usually most effective when coupled with some system that limits access to the A: drive unless the disk has been authorised. For Internet protection it is essential that a scanner be run, on every PC, that will automatically check every executable when it is loaded into memory. This acts as a defence against downloaded software. Often this can only be a limited check so as not to delay program execution for too long, so virus scanners should be used in conjunction with content-based monitors.
- Content-based monitors. These programs usually run on central servers, and the best monitor web downloads and mail traffic. They analyse downloads and e-mail, looking for virus signatures (even in ASCII-encoded binary files) to try to stop these sources of viral infection. In addition content analysis can be carried out, to filter out junk mail and to look for key phrases, providing protection from information leakage (or even libel). Some of these features can be found in firewalls or mail systems, but at least one package provides all of them.
- Token-based access systems. These allow user access to a site based on a token, rather than a simple password. The token typically generates a pseudo-random number, either on demand or every few seconds, which is entered with a memorised PIN. The central server has software that can match the PINs and pseudo-random numbers against individual users and their tokens. Thus the system tries to circumvent the problem of poor password choices whilst protecting against the loss of the token. Historically this technology was aimed at corporate users with dial-in servers that ran classical communications software; naturally, all this technology is now being web-enabled.
- Hidden features and add-ons. Most companies provide features on their products to support security. Often, they are simply not used. For example, the two main PC web browsers can both be set to stop ActiveX and Java components from running: however, these features are rarely activated. The point here is to (1) check all documentation properly, and (2) check providers' web sites: they frequently provide add-ons and extras (and fixes), often free, that improve security.
- Cryptology. This is the subject of the next section.



Cryptography, Cryptology and the Internet

Cryptography is the science of keeping messages secret; cryptanalysis is the science of breaking secret messages. Cryptology embraces both cryptography and cryptanalysis.

The science of cryptography is very ancient: about as old as writing itself. For the definitive history (to about 1960) see [7]; for the currently best modern overall book see [8]. In more recent times it has tended to become the province of the military and governments, and thus shrouded in secrecy, which has led to some unfortunate effects discussed below. Recently, however, a number of effects have come together to open up cryptography to commercial interests:

- The explosive growth of electronic communications since the 1960s, accelerated by the Internet.
- Companies and individuals want to use this communications infrastructure much more extensively. For example, companies wish to use open networks for financial transactions; individuals want to be able to send private details confidentially; organisations want to control their information flows much more precisely.
- Cryptology has developed at an incredible pace since the mid-1970s, when the NBS in the USA first unveiled the proposed DES algorithm at a two day conference (see the Introduction to [9] for some illuminating comments on the state of civilian cryptology at that time).

In effect, society now wants to do a lot more electronically, and wants to do it with adequate security across publicly available communications systems. At the same time, a science has developed to the point where it can meet these needs.

Some of the general benefits of modern cryptology when used effectively are:

- Modern systems have been developed, typically called 'public key systems', that not only reduce the problems of key management, but allow authentication via electronic signatures. Thus people can send messages to each other, without having ever met (or agreed a key), that are completely secure. In addition, a message can be guaranteed to be from a certain person.
- Hard encryption is now easily available. There is a military grade package called PGP that is free and downloadable from the web. This means that absolute confidentiality can be achieved, for both stored files and messages.
- Authentication mechanisms now make repudiation impossible. Having signed a document electronically, the signer cannot then claim it was a forgery; only he or she could have signed it.
- Cryptological functions now exist called 'secure hash functions', or message digests. In concept they are like a CRC, but are typically in the region of 128 to 160 bits long. These can be used to guarantee integrity, and are used in supporting the signature function as well as checking that the message has not been garbled in transit.
- Cryptography can control access. A user must use a key to access encrypted material, so key distribution equals access control. The strength of cryptography is that there is no 'back door': no key, no access.



It should be noted that these are not the limits of modern cryptology. In fact, the field is about the absolute control of information and the level of its disclosure. For example, a number of more esoteric functions and protocols allow the following, amongst many other capabilities (see [8]):

- Demonstrate one has the proof of something, without revealing the proof or even any information about the proof.
- Share a key amongst an arbitrary number of people, so that a subgroup can decrypt a message. For example, a key can be split five ways, but only three people need to get together to rebuild the key to decrypt a message (or open a bank vault).
- Proxy signatures can be produced, so that both you and a proxy signing on your behalf can be absolutely authenticated, and that you gave the power of proxy to the designated person.
- Flip a coin, fairly, over a modem or phone (and by extension play poker by e-mail).
- If a signature key is compromised by a third party by brute force calculation, and is then used to force signatures, these signatures can still be proven to be forgeries.
- Allow partial knowledge exchange. For example, two people can work out who earns more without actually learning each other's income (see [10]).

Perhaps the most active area where the more esoteric protocols are being used is that of electronic cash. This is where one can spend electronic coins that are unforgable, cannot be 'spent' twice, yet are still untraceable back to the spender: a true replacement for notes and coins (unlike credit cards, where every transaction is logged against the cardholder – and the collated information often sold on). There are now two active companies in this field (DigiCash, Mondex), and trials are being run.

Given the breadth of the field and the many cryptographic applications, what is actually being used on the Internet? There are a few basic technologies that are used in many different ways:

- Public key cryptosystems. The best-known system is the RSA system, named after Rivest, Shamir and Adleman, who developed it in 1977 [11]. This system is particularly good as it can support key distribution, encryption and signatures, and is currently the most popular (two other systems that support both encryption and digital signatures are Rabin and ElGamal). Briefly, the system is such that the user has a public key, listed in a database, and a private key, unknown to anyone else. To send a secure message to the user (let us call her Alice), one encrypts the message using Alice's public key. Alice is the only person who can decrypt the message, with her private key. The benefit is that one never has to meet Alice to send her an encrypted message. For authorisation Alice can 'sign' a document by encrypting it with her private key: it can be verified by decrypting it with her public key (if it makes sense, only Alice could have encrypted it). There are two issues:
 - Is Alice's public key really Alice's?
 - RSA is slow.
- The first is handled by digital certificates, where authorities affirm that the key is Alice's, and sign the affirmation with their key. Their key, in turn, can be validated by higher authorities, building a chain of validations until an absolutely unimpeachable validator is reached. These validations are usually called certificates, and standards exist for such certificates (such as



X.509). The second issue is addressed by the use of block or stream ciphers with the application of digital envelopes (see below).

- Because of the slowness of RSA, bulk information is encrypted by a secret key bulk encryption cipher. 'Secret key ciphers' are conventional ciphers: the sender and receiver know a secret key by which the message can be encrypted or decrypted. They are often also called symmetrical ciphers. These are always very fast when compared to RSA, and come in two types: block and stream ciphers. Block ciphers work on blocks of information at a time (often 64 bits, although some ciphers allow for variable blocks); stream ciphers work on one bit at a time. The difference between the two is not so clear in reality, as block ciphers can often be run in single-bit or x-bit modes, where x is variable (such as the Cipher Feedback and Output Feedback modes for DES), and stream ciphers can be made to operate on blocks. It is unknown which type of cipher (block or stream) is theoretically stronger. However, stream ciphers tend to be more efficient in hardware implementations, as the silicon can encrypt each bit as it sees it in a communications system. On the other hand, it is much more efficient for software to encrypt a block at a time. Because it is useful to be able to provide hardware and software encryption systems that can intercommunicate, this means that most practical bulk encryption systems in use are block ciphers (RC4 is an exception). Some well-known bulk cipher systems include:
 - DES: the first major bulk block cipher, produced by IBM for the US Government. Became a FIPS (now NBS) standard in 1977 [12]. Uses a 56-bit key working on 64-bit data blocks. Was finally broken under real-world conditions with unknown plaintext on 17 July 1997 by Rocke Verser and the Internet DESCHALL group.
 - IDEA: proposed by Lai and Massey in 1990 as the PES, changed its name to IDEA (and was improved) in 1992 [13]. Is used in PGP, and may be one of the strongest block ciphers currently available. Uses 128-bit key operating on 64-bit blocks.
 - RC2: proprietary variable key size algorithm designed by Ron Rivest for RSA Data Security Inc. Works on 64-bit blocks. Code is unpublished, so its security has not been greatly analysed and is currently not known.
 - RC5: another Ron Rivest cipher, this time published, with variable block and key sizes (the algorithm itself is also variable in terms of the number of rounds of processing it carries out). RSA Laboratories have spent some time analysing it and clearly believe in it: it is the subject of a set of cracking challenges on the Internet (the 56-bit key variant has been broken in some 265 days by the Bovine Internet group: see www.rsa.com). Will typically operate on 64-bit blocks.
 - Skipjack: the NSA unpublished (secret) cipher system, part of the ill fated US Government EES initiative, which supports 'key escrow' (effectively Government access to cryptographic keys). Based on an 80-bit key working on 64-bit blocks. Skipjack is meant to be implemented on Clipper or Capstone chips. Because of political and technological problems, this will not see much use.
 - Blowfish: a 64-bit block cipher with variable length key (up to 448 bits). Invented by B Schneier [14]. Regarded as fast, seems secure and is appearing in products.



- RC4: proprietary stream cipher invented by Ron Rivest. Was unpublished until the source code was splashed across the Internet in 1994. Seems relatively secure (except possibly for the fact that RC2 and RC4 have special allowable export status from the USA, under certain conditions – which could make the paranoid suspicious). Note that this is the encryption system in Lotus Notes.
- A5: French designed stream cipher used to encrypt GSM mobile phones. Design leaked and now easily available. Based on LFSRs. Mathematics well regarded, but the implementation was clearly cut down to make it not too hard - and (apparently) has been cracked (rumour).
- The third component of modern general-use cryptological technology is a secure hash algorithm. This takes any arbitrary length file or data stream, and compresses it into a known length block (such as 128 bits). The aims are: (1) it should be easy to calculate the hash, (2) it should be infeasible to calculate what document will give what hash result from the hash itself, and (3) given the document and the hash result, it should be infeasible to produce a document with a matching hash. The hash function can be appended to a message, where it acts as a CRC, to ensure that the message is not garbled in transit. In addition, the sender can sign the hash, so confirming that he or she sent it (by encrypting it with their private key: in the real world this is how documents are signed). One extra benefit is that one could send the signed hash without the document, perhaps to claim prior knowledge of something. At a later point one could hash the original message again and resign it to prove that the new and old versions were identical, and so that the document must have existed at a certain point in time. There is the US Government's preferred function: the Secure Hash Algorithm, which is defined in the SHS. This produces a 160-bit hash. Whilst unpopular because the US Government has not published the design decisions (but has published the algorithm), it does seem strong. It will probably become the standard. Another contender is MD5, designed by Ron Rivest. However, concerns are being expressed with this algorithm, and it seems unlikely that it will survive in the long term.

Digital Envelopes

Perhaps the best way to envisage the functioning of all this is to describe a digital envelope. This is a protocol which allows the transmission of messages that are well encrypted. In addition, there is no need for the two parties ever to have met or exchanged keys; furthermore, the receiving party can confirm who sent the message, the sender cannot repudiate the message, and attempts at tampering (or even garbles due to transmission) will be detected. Digital envelopes are a good example of the way modern cryptology is actually implemented, and many modern protocols (even quite esoteric ones) use at least components of the digital envelope.

It will be assumed that RSA is being used for encryption and validation, IDEA for the bulk encryption and MD5 for the hash function (this is, in effect, what the international V2.6.x versions of PGP use). However, the mechanism will work with any appropriate selection of algorithms. Alice wishes to send Bob a message, and they both have published public RSA keys on some central



server. These published keys would be in the form of signed key certificates, with extra information such as Alice's and Bob's mail addresses.

All Alice has to do is prepare the message, and then select Bob as the receiver. The software will then prepare the digital envelope and, in an integrated system, mail it to Bob. What the software does is as follows:

- Randomly generate the IDEA key. This key will only be used for this one message.
- Encrypt this key with Bob's public RSA key. This encrypted key is the first part of the digital envelope.
- The plaintext message is encrypted using the IDEA key. This is the second part of the digital envelope.
- The plaintext message is run through the hash function. The hash function is encrypted using Alice's private RSA key. This is the third part of the digital envelope.
- The three parts of the envelope are sent.

As can be seen, the original message is now embedded in a series of outputs from complex functions, which is why the term "digital envelope" has come to be used. At the receiving end, Bob's software does the following:

- It takes the first part of the received transmission, and decrypts it using Bob's private key: this provides the IDEA key.
- The IDEA key is used to decrypt the bulk of the message, so the original plaintext is recovered.
- This plaintext is then run through the hash function, to generate what may be called the 'received hash'.
- The third part of the received transmission is then decrypted, using Alice's public RSA key. This decrypted hash is compared with the 'received hash'. If they are the same, then the message has been validated, and only Alice could have sent it. If they do not match, then either Alice did not send it, or it has been garbled in transmission.

Note that the digital envelope gives all the benefits of RSA whilst also gaining conventional cipher speeds. RSA is only used to encode 256 bits: the IDEA key and the hash. Everything else uses IDEA. In the real world, there are usually some extra features:

- The keys Alice uses for encryption will probably be separate from the ones she uses for signing. This separation improves management and allows different key lengths for different purposes (the required lifetime of a signing key is likely to be much greater than that of an enciphering key: consider your mortgage).
- Other information is likely to be included in the message header, such as date stamps and key identifiers so that the software can find the appropriate public keys for decryption.
- Validation certificates may well be incorporated into the message (to validate the signature of the sender, especially if the availability of key servers is limited).
- The plaintext is likely to be compressed first (by some algorithm like PKZIP), simply to reduce size and redundancy.



- The whole digital envelope is likely to be converted into ASCII characters for Internet transmission (via some program like UUENCODE).

One important point to note is that it is not essential to have databases of public keys for the system to work. It is likely to be safer, because usually the database would require validation certificates before it would include the keys. However, Bob and Alice could simply send each other their public keys. They could then run the protocol just as effectively as before. But they would be more open to attacks such as the 'man in the middle' attack.

Cryptographic Use and Its Problems

The discussion above indicates the basic functioning of cryptography, and it is clear how the digital envelope can be used for any messaging capability. Increasingly, mail systems are beginning to incorporate such technologies (such as S/MIME and PEM), and PGP has been available for several years. Lotus Notes also incorporates an encryption system based on RC4, along with digital envelopes that include RSA for key exchange and authentication certificates.

Network operating systems are developing support for encryption at the packet level; for example, NT supports data encryption for dial-in lines using RC4. Local LANs are typically not encrypted, but this is changing as organisations tighten their security.

Of course, not all of the components of the digital envelope are required for different functions. For example, file security applications for storage on the local hard disk may well only use the secret-key algorithms such as IDEA, as that is all that is required. Cryptography is now increasingly being used to protect information on disk, which adds an extra layer of security for both servers (that may be attacked by hackers) and notebooks (that can be stolen).

Modern browser systems support RSA-based key exchange protocols, where public keys are exchanged so that a private session key can be negotiated. The SSL protocol created by Netscape, for example, is now widely available, and allows secure information transmission between the server and the browser: the protocol is certainly good enough for the transmission of credit card numbers in its design. In addition there is PCT, developed by Microsoft, which is claimed to be an advance on SSL.

The other area where the protocols are being heavily used is in authentication. Obvious areas include password authentication and key negotiation, and signing mail messages. In addition, entire protocol suites such as SET are being developed to support credit card transactions over the net.

However, authentication is being developed for much wider purposes, such as software verification. Microsoft are supporting the VeriSign Authenticode™ system, where net software has certificates attached. These signed certificates show the creators of the software, as authenticated by VeriSign. Thus when a signed applet is downloaded into the browser over the web, both the



identity of the author can be checked and the fact that the code has not been altered since creation: a significant enhancement to software security. In the case of Java, such capabilities are actually part of the original design specification of the language, although they have yet to be implemented.

Whilst use of, and interest in, cryptography are increasing, there have been factors that have seriously retarded the development and application of cryptography. Some of the reasons are:

- A lack of standards or, from another point of view, too many of them. Because this science is undergoing a rapid rate of evolution, no set of products or protocols has gained a significant share of the market. From the point of view of basic algorithms and protocols there are:
 - Three ways of using public key cryptography by public/private keys (RSA, ElGamal and Rabin) plus new ways of implementing these using elliptic curve mathematics. There is also the original public key exchange protocol (Diffie-Hellman) which is in use. The US Government has gone for a digital signature system that is incompatible with all of the above (DSA), which undermines the use of other systems for authentication.
 - An extraordinarily large number of bulk encryption systems. The more common or well-known ones are: DES, 3-DES, IDEA, RC2, RC4, RC5, Blowfish, A5, Skipjack/Clipper. These are all in use (even if the Skipjack community is very small). [8] Refers to at least 30 block ciphers in the Contents, and new ones are appearing all the time.
 - At least two popular hash algorithms (MD5, SHA), and others are available.
- At the higher levels, products cannot intercommunicate. For example, the following different mail-supporting encryption standards exist: S/MIME, MOSS and PEM (plus MSP as a military version of PEM). None of these standards are interoperable; worse, they are all still technically proposals (although PEM is virtually ratified). Most current mail products are proprietary (almost by definition, without standards), obviously limiting the usage of cryptography.
- Another issue is the general lack of knowledge about encryption, and the benefits it can produce. This is not helped by scare stories in the press, often unclear if not mistaken, about security holes in cryptographic products.
- Encryption is often perceived to be expensive. Interestingly, many cryptographic utilities and products are effectively freeware of the Internet (such as PGP 2.6.x, international version). There are costs associated with cryptography: mostly in terms of its key management. Most organisations have no experience in this area, and so find it difficult to budget for this.
- One of the major benefits of cryptography is the capability of authorisation and accreditation. However, for this to be effective there is a need for a global certification system of some sort, so that people's identities can be validated and 'bound' to their electronic signatures. This is so that one has some confidence that the person who is signing an electronic bank draft is the person he or she claims to be. No such certification infrastructure exists, and it is difficult to see the real benefits of cryptography being gained until one does.
- Another issue is Governmental attitudes towards cryptography.



Cryptography and Governments

Governments vary in their views on the use of cryptography. Totalitarian regimes usually regard any use of it as de facto evidence of subversion, but excluding these there are quite extreme ranges of view. Switzerland and Japan, for example, have effectively no controls on its use of the selling of products. France, on the other hand, makes the use of strong encryption virtually illegal. A licence is required, and no one has been given a licence to use PGP, for example (this is expected to change in 1998, but the position is still not clear).

The UK is relatively tolerant. Currently there is no control on usage within the UK, and there are some limited controls where a licence is needed for export to certain countries. However, there has been a consultation document put out by the last Government [15] which raised the concept of 'key recovery' (an upmarket term for key escrow, which means centralised key storage and access by Government bodies) through the medium of trusted third parties (TTPs). These licensed agencies would store encryption keys, and the Government would, via Court Orders, have access to them. Any organisation providing an 'encryption service' (with a few exceptions, such as satellite decoder and selected bank services providers) would need to be licensed.

This document met a lot of opposition, and a large number of comments were received by the DTA within the 30 May deadline (which was an extremely short consultation period given that the proposed legislation could have fundamentally removed any right to privacy or confidentiality). To be fair, the proposal did try to address some of the concerns outlined above, such as the creation of an accreditation structure and dealing with the international issues. It also stated that it was not looking to criminalise the private use of encryption and that the use of TTPs could be voluntary (although paragraph 47 undermined such assurances by reference to future legislation). However, the document was flawed in a number of ways, as a couple of examples will show. First, a number of proposals were unworkable, such as the time given for yielding up a key and the international proposals. Second, there seemed to be misunderstandings about the technology (for example, the document did not seem to envision the case that to monitor a suspect using RSA-based cryptography, access is needed to the private key of every single person he or she contacts).

Since the publication of the document, there has been a change in Government, and the Labour Party has so far indicated that it will be more open on the subject of cryptology. What is clear is that the initiative started by the last Government has come to an at least temporary pause, and there may well be a new cycle of consultation in the future. However, it is interesting to note that the Conservative Party was moving strongly towards Governmental control of cryptography, the fundamental justification being that criminal groups and terrorists need to be controlled. This justification is, to say the least, arguable. Its importance lies in the fact that it is the justification for the US Government's position on cryptography, which has probably had the greatest negative effect on the development of cryptography of all.

[As of current date (31 March 1998) it is rumoured that the UK government will publish some new proposals. What these are is not known.]



The US Government defines strong cryptography as 'munitions'. Export has been tightly controlled. In addition, whilst the US Government allows strong cryptography to be used domestically, it does not like it. It has thus tried on more than one occasion to limit it. In 1993 the US Government announced the Clipper and Capstone proposals for the Escrowed Encryption Standard: a formal proposal to allow access to every cryptographic key in use by forcing the usage of the Clipper and Capstone chips. These carry an implementation of the Skipjack algorithm, which allows access by law enforcement agencies to the key by a relatively complex process. This involves two key storage organisations that each have half of the key: the escrow agencies (the Treasury and NIST, both part of the Government Executive).

Whilst access was to be via Court Order, many people believed that the whole proposal was an attempt to remove a right to privacy by the Government. Since then, the whole Skipjack escrow system has been shown to be fatally flawed [16], allowing it to be circumvented. Skipjack, and its Clipper and Capstone implementations, is now effectively dead and has to all intents and purposes been withdrawn (by 1996 a total of 10,000 Clipper chips had been sold [11]).

The US Government then attempted to get a Bill through which attempted to achieve the same ends by legislation: all cryptographic keys needed to be registered with 'Trusted Third Parties', which would yield the keys subject to a Court Order. This was referred to as 'Clipper II', and died in the Senate in 1996. The US Government is currently trying again, with another Bill (the McCain-Kerrey Bill). It is effectively Clipper II, but only makes key registration mandatory for Government agencies and anyone who wants to do business with them, and for financial transactions. In practice this would effectively force everyone to use the TTP/escrow system.

Thus the US intends to control encryption domestically. Alongside these attempts, it has been successfully stopping American companies from exporting strong cryptography. Typically, the only things that have been allowed to be exported have been products incorporating RC2 or RC4 with limited 40-bit keys, well known to be breakable in very short periods of time. Browsers have also been similarly limited. DES with 56-bit keys needed a special licence, usually only granted to large American companies with foreign subsidiaries or large multinational banks.

These policies of the US Government have had major effects on encryption and its development:

- The limitation of exports has severely damaged the cryptographic industry. It created a watershed between America and the rest of the world. Most software companies are in the USA, and they have been limited because they cannot export their products. Whilst this situation might favour non-US companies, this benefit is only limited because they cannot really sell into the US market. The Microsofts and Lotuses have already developed their products, they just cannot sell them overseas.
- The US attempts to control domestic US encryption have destabilised the market for two reasons. First, it was not clear if cryptological technologies would be criminalised. Second, if they were legal, it was not clear if the Government would foist some new, non-standard system



onto the community (such as Skipjack, and it is notable that every NSA algorithm the Government has implemented, such as DSA and SHA, have been non-standard).

When the mixed views of other Governments is added to those of the US Government, it is clear why cryptography has not taken off the way it should. Currently, it would be extremely difficult for a multinational to set up a global encrypted network that was both strong and legal in every country. This is very unfortunate, because all our lives are becoming more and more dependent on, and immersed in, the vast electronic communications infrastructure of this planet. It is becoming increasingly important that we have secure and confidential information in this environment.

On a hopeful note, there are indications that the situation may improve. The US Government, with the failure of Clipper II, is beginning to allow companies to export stronger encryption on the understanding that they will develop key escrow technologies within two years. The OECD has published cryptographic guidelines that seen a sensible compromise between the privacy of the individual and the needs of society [17]. Cryptography has the potential to provide excellent security for all of us in the new electronic society.



Definitions and Abbreviations

This list of definitions and abbreviations aims to support the concepts covered in the talk. It is neither exhaustive nor definitive; it represents the author's view.

Cracker	Someone who breaks into systems for profit
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DSS	Digital Signature Standard
EES	Escrowed Encryption Standard
GSM	Group Special Mobile
IDEA	International Data Encryption Algorithm
IP	Information Protection
ISP	Internet Service Provider
Hacker	Someone who breaks into systems for fun or fame
LFSR	Linear Shift Feedback Register
Logic bomb	A program that causes a destructive event, activated by a trigger
MSP	Message Security Protocol
MOSS	MIME Object Security Standard
NSA	National Security Agency
PCT	Private Communications Technology
PEM	Privacy-Enhanced Mail
PES	Proposed Encryption Standard
PIN	Personal Identification Number
Payload	The executable part of a virus that is not involved with reproduction; its delivery (a message or action)
RSADSI	RSA Data Security Inc
SET	Secure Electronic Transaction
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
S/MIME	Secure/Multipurpose Internet Mail Extensions
Spamming	sending junk E-mail all over the Internet, to (very) large numbers of recipients
SSL	Secure Sockets Layer
Trojan horse	Program that claims benefits, but has damaging effects by design
TTP	Trusted Third Party
Virus	A program that uses host resources to reproduce itself, and may contain a payload
VPN	Virtual Private Network (isolated networks that are joined by encrypted links to be a single logical network)
Worm	A program that reproduces itself without using other program files, and typically spreads through networks via mail systems
WWW	World Wide Web



References

- [1] *Keeping an eye on....security*, p19, Network Week, 25 June 1997.
- [2] *Information Security Management*, British Standard BS 7799:1995.
- [3] *News*, Secure Computing, p8, January 1997.
- [4] *Where is our data?*, Secure Computing, p22, August 1997.
- [5] *News*, Secure Computing, p9, June 1997.
- [6] *Muse: Use It or Lose It*, p14, Secure Computing, May 1997.
- [7] *The Codebreakers*, David Kahn, Macmillan, 1967.
- [8] *Applied Cryptography*, B Schneier, John Wiley & Sons Inc, 1994, 1996. Note: new editions come out periodically.
- [9] *Machine Cryptography and Modern Cryptanalysis*, C A Deavours and Louis Kruh, Artech House, 1985.
- [10] *Protocols for Secure Computations*, A C Yao, IEEE, 1982.
- [11] *Answers to Frequently Asked Questions About Today's Cryptography*, V3.0, P Fahn, RSA Laboratories. www.rsa.com/rsalabs, 1996.
- [12] *Data Encryption Standard*, FIBS PUB 46-1, 1977.
- [13] *A Proposal for a New Block Encryption Standard*, X Lai and J Massey, p389, EUROCRYPT 1990, Springer-Verlag.
- [14] *Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)*, B Schneier, p191, Fast Software Encryption, Cambridge Security Workshop Proceedings, Springer-Verlag, 1994.
- [15] *Licensing of Trusted Third Parties for the Provision of Encryption Services*, Public Consultation Paper on Detailed Proposals for Legislation, DTI, March 1997.
- [16] *Protocol Failure in the Escrowed Encryption Standard*, 2nd ACM Conference on Computer and Communications Security, M Blaze, p59, ACM Press, 1994.
- [17] *Cryptographic Policy Guidelines, Recommendation of the Council*, OECD, www.oecd.org, 27 March 1997.