Protecting IT Systems from Cyber Crime

R. Benjamin
Visiting Professor: Imperial College, London;
University College, London; University of Bristol.
13 Bellhouse Walk, Kingsweston,
Bristol BS11 OUE

B. Gladman
Independent consultant, specialising in Information Security.
4 Palmers Green, St Johns,
Worcester WR2 4JY

B. Randell
Professor of Computing Science.
Department of Computing Science, University of Newcastle,
Newcastle upon Tyne,
NE1 7RU, UK

Abstract.

Large-scale commercial, industrial and financial operations are becoming ever more interdependent, and ever more dependent on IT. At the same time, the rapidly growing interconnectivity of IT systems, and the convergence of their technology towards industry-standard hardware and software components and sub-systems, renders these IT systems increasingly vulnerable to malicious attack. This paper is aimed particularly at readers concerned with major systems employed in medium to large commercial or industrial enterprises. It examines the nature and significance of the various potential attacks, and surveys the defence options available. It concludes that IT owners need to think of the threat in more global terms, and to give a new focus and priority to their defence. Prompt action can ensure a major improvement in IT resilience at a modest marginal cost, both in terms of finance and in terms of *normal* IT operation.

Index Terms: Security, Dependability, Cryptography, Networked Systems, Crime Protection

1. Introduction

1.1. Background

Industry, government and indeed society are becoming critically dependent on IT [16; 18]. This dependence is illustrated by the serious concerns which are now being caused by residual “Year 2000” bugs. Seeing that even these conceptually-simple software faults are demanding massive resources, we must be concerned about the much more difficult effects of “cyber crimes": malicious activities by “hackers" or organisations seeking to exploit or disrupt an IT system, for mischief, financial gain, or more sinister motives [23].

Such cyber crimes cannot be considered separately for individual systems, because of the rapidly growing interconnectivity between IT systems, via Intra-nets, Extra-nets and the Internet itself, as well as by direct physical interconnection, or exchangeable storage media such as diskettes. Such interconnectivity (often unintended, rarely adequately planned) turns separate IT systems into components of what is in effect a single large super-system that might suffer an overall failure, or whose data or software may be seriously polluted as a result of a single malicious act (or accident). Indeed, with the growth of electronic funds transfer, just-in-time procurement, concurrent engineering, etc., almost any organisation's or company's IT Infrastructure is in danger of becoming an ill-defined part of an ill-defined global system, whose configuration and operation are not under any effective overall control.

The increasing use of standard interfaces and protocols has provided major advantages for the user community. At the same time, however, this facilitates the initial access for an attacker. The widespread use of virtually standard data-bases, spread-sheets and other generic software applications and components, and of standard hardware processors, together with the continuing evolution and dissemination of hacking tools and techniques, makes the attacker's subsequent deeper penetration into our IT systems ever easier. Furthermore, such attacks are difficult to detect and harder to trace to their source, and the hacker can work from a location where he (or she?) is essentially safe from legal retribution, thus making such attacks ever more tempting.

For example, it has been reported in the USA that there were at least 250,000 attacks on Department of Defense computers in 1995. (Possibly there were many more: estimates are difficult, since many attempts are undiscovered and many others, albeit discovered, are not reported.) Of the known attacks 65% were successful, and the number of attacks is doubling each year. “Attackers have seized control of entire Defense systems, many of which support critical functions, such as weapons systems research and development, logistics, and finance. Attackers have also stolen, modified, and destroyed data and software.” [10]. Reported attacks on commercial systems include one by a group of hackers who were caught and convicted of breaking into computers of Bank of America, South-western Bell, Martin Marietta, and TRW Information Services, and selling information, such as credit reports, that they so obtained [15]. In the UK there has been a defensive tendency to downplay or deny the seriousness of cyber crime, and to keep quiet about embarrassing occurrences. Even so, there have been enough publicly admitted incidents to make it clear that cyber crime threatens us all. (For example it has been claimed that UK banks have suffered losses amounting to £500 million through ransom payments or having to suspend activity as a result of hacking attacks [14].)

The potential seriousness of cyber crime is even greater if it affects critical IT systems of telecommunications, power distribution, banking or transport, i.e. of the infrastructure on which virtually all individual companies depend. Such concerns led the US President to set up a Commission on Critical Infrastructures [21]. However, in this paper we deal solely with the defence of corporate IT systems.

Some major IT systems are of such evident critical importance to the organisations that own them that these systems employ hardware and sometimes also software redundancy to ensure system availability, use encryption to maintain information integrity and confidentiality, are operated under careful management, and have well thought-out disaster recovery plans for situations when major failures do nevertheless occur. However, the main concern is usually accidental hardware or software faults, or operational errors. Few civil organisations have devoted much effort to defence against deliberate cyber threats. However, this threat is real and growing, and there are useful steps which can and should be taken to provide improved defence against it.

This general survey paper focusses on computer/IT systems, and is aimed particularly at readers concerned with major systems employed in medium to large commercial or industrial enterprises. It is a sequel to [5], commissioned by the IEE 10 years ago, which gave a comprehensive overview of the role of cryptographic devices and concepts in the design of communications systems: to protect their information, authenticate their subscribers and messages; and prevent, impede or at least detect various forms of disruptive or deceptive interference with network operation. Other papers, such as [8; 22], treat design principles to protect IT systems against accidental - or maliciously caused - hardware or software failure, and much effort continues to be devoted to testing and verification techniques for both hardware [11; 13] and software [4; 19]. A few articles, such as [6], also address the management and operation of systems, so as to realise and maintain the potential security benefits of these design features. All these publications focus primarily on the protection of an identifiable system or network, even though this may include many and diverse subsystems or subnets.

However, industrial, commercial and administrative processes and IT systems are becoming more and more interdependent:-

> ***Any equipment, system or activity is at risk, if it depends, directly or indirectly, on any other equipment, system or activity which depends on computers or communications for its operation, control monitoring or management.***

Thus a system may be at risk due to a technical or procedural failure, an accident, or a malicious act **affecting someone else's system**. Conversely, in defining, designing and operating our own systems, we have to be aware of our moral responsibility to all those other people, organisations and operations who might suffer incidentally, as a result of an IT failure in our system. In this paper, we discuss the risks to IT systems, and ways to contain these risks. Since robustness in the face of technical failure is already receiving considerable attention, this paper deals with the risks of malicious attack on IT systems, and ways to detect and cope with, or to prevent, such attacks. (The planned revision of BS 7799, the widely accepted standard for information security, gives this topic extra urgency.)

1.2. The Present Situation.

In commerce, industry and the public utilities there are now numerous systems of systems - which, ***as regards their security features,***:

- are a haphazard collection of disparate, poorly-structured sub-systems,
- have evolved in an unplanned manner,
- and are often used and managed with little regard to their security.

On the other hand, most of our industrial IT infrastructure is still sufficiently fragmented that there remains a window of opportunity to guide its evolution towards improved security through the progressive introduction of components, such as interface controllers, that provide more effective defences in the face of hostile attack. When properly implemented and managed, such interface controllers (guards, gateways and firewalls) can greatly enhance the security of systems involving the following classes of data flow - particularly where these do not already benefit from end-to-end encryption:-

• Those flowing “transparently” across a constituent system or network,
• Those traversing interfaces between systems and human or processor “users”,
• Those at interfaces between systems or subsystems.

TABLE 1: Signal/Data flows benefiting from interface controllers.

As discussed in section 6, these interface controllers are a crucial tool, but their design, placement and management involves financial and operational trade-offs, and they are no panacea. On the other hand, there are also practical steps for enhancing the security *within* existing systems or sub-systems. The scope for some of these options may be restricted by past decisions, but much can still be done, often at relatively modest cost. Certainly little cost is involved in producing quite effective user-friendly security monitoring and control interfaces and good security-training material, and to inculcate users and managers with an awareness of their security responsibilities.

In the following sections we discuss the principal threats and possible counters to them. Whilst some distinction is made between attacks on communications and computers, we note that the nodal switches and the communications network-controllers and management systems are computers, distributed computer systems are interconnected by communications networks, and indeed single computers commonly rely on communications for many of their inputs and outputs, and even the “buses" interconnecting their modules are basically local-area networks.

Sections 2 - 5 deal with various “passive" and “active" attacks on communications and the corresponding defences. Sections 6 - 9 cover the same subject in relation to computers, sections 10 - 16 deal with management and policy aspects of IT security, and section 17 draws conclusions.

2. Unauthorised Interception

Interception of communications is normally undetectable and, in the absence of suitable countermeasures, offers a tempting target to attackers. In appropriate computer systems, unauthorised access to data-bases, etc., *can* be monitored and, where this has been done, it has produced ample evidence that probing attacks are indeed taking place on a substantial and increasing scale. In the present context we regard all attacks which solely seek to gain information, from communications or computers, as *“passive”*. We distinguish five types of such passive attacks - see TABLE 2:

Type of attack	Comment
random (fishing) attack on message or file content	• random attack on plain text is quick and easy • countered by encryption
“Space-Domain” analysis of organisational structure	• needs sustained effort, but attack is otherwise easy • difficult to counter • **pre-requisite for the three forms of attack listed below**
“Time-Domain” traffic-flow analysis	• needs sustained effort, but attack is otherwise easy • a little harder still to counter
message interception or data extraction targeted onto key parts of the system	• must be countered by high-grade encryption
systematic, system-wide message interception or data hacking	• needs substantial, sustained high-grade effort by attacker • must be countered by high-grade encryption

TABLE 2: Passive Attacks

Whilst, in the absence of suitable defensive measures - see below - a fortuitous probing attack can often produce some results within minutes or even seconds, the final three classes of passive attacks, when directed at a large, complex system or network, may need a prolonged “reconnaissance” and analysis of relevant parts of the system's architecture. Similarly, attacks to produce a major disruption of service - discussed in later sections - will generally have to be preceded by a thorough “reconnaissance” (except for self-spreading “virus” infections, if not limited by secure interfaces).

The principal defence options are listed briefly below:

Defence against interception of trunk links:

- Bulk encryption. This can normally be implemented at relatively low cost, even retrospectively, and should be a standard feature in all networks carrying identifiable critical traffic.

Defence against attack on decrypted redistribution nodes:

- Physical & personnel security. This should be - and mostly is - a “matter-of-course” for all significant facilities.
- Routine end-to-end encryption of selected virtual links. This entails the creation of an appropriate key-distribution system - which may use “public-key cryptography” - but it can then be superposed in a “transparent” manner on a network - including its encrypted trunk links (i.e. super-encryption) - for the duration of relevant connection session. It should be a standard facility for traffic or users of particular criticality or sensitivity. (The segregation of signalling in a separate channel facilitates this, but increases vulnerability to some other forms of attack.)

Protection of sensitive connection sessions:

- End-to-end (super)-encryption. This is a variant of the preceding technique, applying it only to those connection sessions judged to warrant it

Defence against Analysis of Organisational Structure:

- Masking differences in traffic type or volume between different organisational entities.
- Avoidance of distinctive emissions.
- Defence against traffic-flow analysis (see below).
- Using interface controllers to hide internal systems and network configurations

It is frequently worth-while to make structure analysis harder for the attacker, but rarely practical to make it impossible.

Defences against traffic-flow analysis.

In some critical situations, in public or corporate governance, the mere existence of a very non-standard pattern of traffic flow could give away sensitive information. If necessary, we may impede or prevent such analysis by:

- Dummy traffic.

- Dummy “post-box” addressees - interface controllers can sometimes also subsume this function.
- Filling non-busy links with “empty” key stream.
- Encryption of trunk signalling channels.

This would be warranted only for a few, particularly sensitive, types of traffic.

Diversion and Deception

Another defence against passive attack is that of diverting the energies of hackers into ‘dummy’ sub-systems of no strategic value but which are designed to appear attractive to those who are attempting to penetrate the systems concerned. Simple examples of such strategies are the ‘dummy’ password files now often seen on systems subject to external access. At a more sophisticated level, some critical Internet-connected organisations have established significant ‘dummy’ resources for such diversionary purposes. Such approaches can be costly, but they also have the advantage of encouraging an attack to continue so that it can be analysed without putting real data at risk.

3. Interference with Communications

Various forms of interference with communications may also be encountered, either ‘merely’ aiming to cause mayhem or for fraud or blackmail. All cause the receipt of incorrect messages or prevent the receipt of the correct messages. The principal forms of such “*active*” attacks are listed in TABLE 3.

causing delivery of incorrect messages	• false retransmission of true messages • diversion of true messages • false attribution of true messages • generation of “spoof” messages
interference with message content	• cropping • modification
interference with network	• saturation • “fouling-up” switches • “fouling up” the network management system
interference with synchronisation (mainly in radio systems)	• attacking message synchronisation • attacking crypto synchronisation

TABLE 3: Active Attacks

Some of these attacks may be limited in scope, e.g. to a single communications link or a single local computer or a single type of transaction, or they may be limited in the duration of their impact. On the other hand, attacks which immobilise network-control or destroy confidence in the traffic carried would have a very grave pervasive impact, until countered, and so resilience against such attacks is of high importance. As far as we know, none of these active attacks have yet been deployed for criminal, political or terrorist purposes. However, all have been discussed in the literature, and the vulnerability of many important systems to such attacks has been proven by “tiger teams” emulating potential attackers. As shown below, most of these active attacks can be *prevented* by the appropriate use of crypto techniques, coupled with good, active security management. However some (such as interference with synchronisation) can merely be *detected* and made more difficult, and others (such as partial or complete deletion of messages) can only be detected.

Defence against malicious retransmission of formerly valid messages:

- Time-dependent crypto key - or at least encrypted time stamp and serial number.

Where the end-to-end links are already encrypted, this entails no extra cost.

Defence against receipt of incorrect message, due to diversion or false attribution of true messages, or to masquerading to send “spoof” messages:

- Authenticated end-to-end session crypto keys.

Where the end-to-end links are already encrypted, backed by an appropriate key-distribution system, this entails no extra cost.

Defence against malicious system saturation:

- Detection and rejection of spoof messages (see above).
- Limitations on traffic accepted:
  - from any one source,
  - and/or to any one destination,
  - and/or on any one link.

This entails the use of a trusted gateway but, where this is used for its wider security benefits, it involves no further cost.

Detection of deletion or diversion of message:

- Encrypted serial numbering plus minimum frequency of channel-testing messages.

Where the end-to-end links are already encrypted, this entails no extra cost other than that of introducing the test-message generation protocol.

Detection of message cropping or modification:

- Encrypted parity checks or “digests”.

No problem (other than the parity processing) if the connection is encrypted.

Jamming of radio systems is a well-understood threat, and we do not discuss it here. However, interference with critical synchronisation functions could be a more subtle and much more potent threat, equivalent in its effect to jamming, and this we touch upon below:-

Defence against interference with message synchronisation (in radio or satellite links):

- Robust synchronisation sequences

  This is normal good design practice. It makes the attack more difficult and less effective, but offers no complete protection.

- Crypto concealing message synchronisation.
  At the expense of transmitting an “empty” key stream at non-busy times, this can prevent any attack focused specifically on synchronisation. As mentioned earlier, it also maintains traffic-flow security.

Defence against interference with crypto synchronisation (in radio or satellite links):

- Robust crypto synchronisation sequences.
  This is normal good design practice. It reduces the vulnerability, but does not constitute a true defence.
- Use of block codes - which depend only on *message* synchronisation, (see above).
- Use of “key-autokey” key generators, which minimise error propagation. In the present context, this, too, comes under the heading of “good design practice, but offers only a limited defence”.
- Maintenance of a steady key-stream, in the radio link, even in the absence of traffic, masking synchronisation of any end-to-end super-encrypted connection.

The likelihood and impact of these various attacks will guide the choice of crypto architecture, but all crypto systems should include the simple, low-cost protective functions listed above.

Protection of network operation and management

- Isolate signalling traffic from user traffic
- Isolate operation and configuration management traffic from signalling traffic.

The options for such isolation are listed in Table 4 (by courtesy of M. Ward [24]):

- Separate parts of same packet (e.g. “connectionless” operation),
- Separate packets in same stream (e.g. Asynchronous Transfer Mode),
- Separate streams in same wavelength (e.g. Synchronous Digital Hierarchy),
- Separate wavelength in same fibre,
- Separate fibre in same cable,
- Separate cable in same duct,
- Separate ducts in same route,
- Route diversity.

Table 4: Degrees of Isolation of Functions

4. Physical Attacks Against the Bearer Net

We also have to consider attacks against the physical bearer net. Terrestrial networks have at least the *potential* for a very high degree of redundancy in their capacity and topology/cross-connectivity. The proper exploitation of this potential robustness requires:

1. Monitoring the availability and capacity of the various links,
2. Adaptive routing,

3. Prioritisation,
4. Possibly adaptation of operational procedures to tailor traffic load to the capacity available,
5. If necessary, restriction of access rights for lower-priority traffic.

Facilities 1 and 2 are normally quite good. Crude versions of 3 and possibly also of 5 exist in many networks. Facility 4, a potentially powerful form of adaptation to network damage, is insufficiently planned and practised. “Fixed” telecommunications networks are generally sufficiently redundant to be quite resilient to damage to trunks or nodes - or to jamming of terrestrial or satellite radio links. Subject to any capacity impairment by jamming, satellite links can also make a major contribution to network reconstitution or reconfiguration. Unfortunately, it is not uncommon for functionally independent trunks to be physically collocated (for at least part of their route), or for switches to be accommodated side-by side in the same building. However, contractual or financial incentives might persuade communications operators to adopt more defensible dispositions.

A low-capacity backbone network of assured survivability is typically required, both to control the reconfiguration and restitution of the main communications infrastructure and, pending such action, to provide an assured route for a minimum of vital operational communications.

Electro-magnetic pulse (EMP), be it due to an air-burst nuclear explosion or a dedicated device, can only be produced by a determined and sophisticated, well-resourced attacker. Fibre *links* are immune to EMP. Where *terminal equipment* is designed not to produce emissions which could interfere with nearby sensitive equipment or which could be intercepted and exploited by an interceptor, it needs little further care in design to make them also resistant to EMP.
(Any connection which:

- penetrates the screened box,
- is not optically coupled,
- is connected to a structure which could act as a pick-up antenna,

would have to be fitted with a peak-power limiting or bypass device.)

Where a particularly sensitive system is already designed for minimum emission and good EMC resilience, it may entail little extra cost to make it also EMP-safe, and it seems then advisable to do so. Beyond this, only a small core of vital facilities and connections may warrant dedicated EMP protection.

5. Attacks Against Crypto Systems

Crypto systems play a critical role in maintaining the security of IT systems against unauthorised reception and exploitation. Furthermore, they are also key components in:

- authentication,
- preventing various subtle attempts to pervert the communications process,

- security in data bases,
- imposing security restrictions on inputs to, outputs from, or interactions between computer hardware or software modules.

Thus their very ubiquitous application and importance could make them tempting targets for attack.

Various forms of attack can be envisaged, involving decryption, physical interference with crypto equipment, or interference with key management or distribution. All of these can be countered by the combination of means listed in TABLE 5:

- Sound COMSEC (COMmunications SECurity) design,
- Safe key distribution and management,
- Simple and well-conceived COMSEC operating procedures,
- Good operational COMSEC discipline,
- Automatic alarms when equipment or humans perform incorrectly,
- Monitoring of the compliance with those procedures which cannot be covered adequately by automatic alarms.
- Prompt and decisive action in response to any defects so discovered.

TABLE 5: Requirements for a secure crypto system

The security of high-grade encryption-systems does not depend on secrecy of the crypto algorithm, but merely on secrecy of the key used - the decryption key, in the case of public-key cryptography. However, in the design of sophisticated cryptographic algorithms and security protocols, is it is easy to make subtle mistakes - which then can have a disastrous impact on the security achieved. Hence is it essential to subject such algorithms and protocols to extensive review and scrutiny and test, before bringing them into use. Historically, in many custom systems designs, such scrutiny has been missing, often leaving major weaknesses undetected [1].

In public key cryptography, anyone can encode "a message" of his own choice by the public encryption key, and can then use brute force to try all possible decryption keys. Such systems had been regarded as secure for, say, the next ten years if the extrapolated growth in the power of computers and predicted "upper limits" to likely improvements in attack algorithms imply that, even with the most powerful computers and the best attack techniques that might become available by 2008, no effective cipher-breaking effort would become practical. However, we are faced not only with the continued exponential growth in performance of individual computer systems, but also with the growing proliferation and interconnectivity of computers. Indeed, the ability to combine thousands of computers temporarily, and thus to break such a key, has already been demonstrated. If the "borrowed" computers can be organised in a two-level hierarchy, the use of even millions of computers could be envisaged, in the future. Hence life-time predictions will have to take into account the extrapolation of the growth of *global* computer power. On the other hand, even such breaking of the decryption key corresponding to a public key becomes a practical threat only if the attacker can focus his collection effort on traffic to the relevant target, or at any rare on a traffic stream which includes a significant proportion of such traffic. This may well be frustrated by the growing volume of global traffic, and by the various system-wide COMSEC measures, impeding selective traffic collection, discussed towards the end of this section.

Security is substantially better with more conventional "private key cryptography". Here an attacker has to rely on intercepted and identified relevant traffic, rather than self-generated cipher text. Moreover, he does not know the underlying plain text, and this makes the success criteria of an exhaustion attack substantially more difficult, and makes such attacks much less amenable to the use of "borrowed" computer resources.

However, a much more dramatic gain still, in security, is obtained when a "subscriber unique" key, known only to the given subscriber and the key-distribution centre, is used to assign to him a "session key", used only for a single message or transaction or a limited-duration connection. Thus, unlike the breaking of a long-duration public or private-key crypto system, any exhaustion attack, if viable at all, is then a threat only to a single isolated item, and then only if and when the critical terminal and link, and the critical item, and the correct moment for collecting that item are known to the would-be cyber criminal and exploitable by him. Otherwise he is looking for a small needle in a very large haystack, in trying to find the item that he wants to try to decrypt. Indeed, the attacker may be looking for a small *hay-fibre* in a large haystack, if he is faced with one or more of the following likely COMSEC developments:-

- trunks are bulk-encrypted
- signalling channels are encrypted
- addresses, routing indicators and location registers are encrypted or otherwise not accessible to the attacker
- a substantial proportion of traffic is end-to-end encrypted (not necessarily to high standards).

This emphasises the complementary nature of all aspects of system security architecture and policy. Utmost care is needed to deny the attacker the chance of focused data collection at critical terminals and links and, throughout the total network or system, we must deny him access to any easy discriminant, which might help him to pick the nuggets from the dirt.

Even the security of a basically sound design schema can be undermined by the fortuitous introduction of exploitable weaknesses during implementation. Furthermore, many security protocols are currently implemented in packaged commercial software which is not designed - and indeed probably cannot be designed - to the appropriate level of assurance. Hence, for high-level security, such protocols will need dedicated hardware. Even then, tampering with the crypto device may not be beyond the capability of a small - albeit high-calibre - electronics facility [2; 3], and hence good physical security is essential.

Needless to say, the COMSEC architecture and policy should, wherever possible, be an integral part of system design *ab initio*, and good COMSEC discipline and management must be an integral part of subsequent system operation.

6. Software Attacks on Computers

Much of the most vulnerable traffic flow is to, from or between computers. Computer systems generally comprise discrete hardware and software modules, intercommunicating via “buses” or networks. Hence their defence has much in common with that of communications systems. However, even more than in Communications, most computer hardware and software is produced with no close control over its security features and, moreover, computers are vulnerable to self-propagating “virus” or other infections. Furthermore, attacking an allegedly “secure” computer system represents the same sort of challenge to to-day's youngsters as climbing an “unclimbable" mountain-face - but with far more opportunities to make the attempt - and much smaller risk:- There is almost unlimited scope for placing space, time and logic cut-outs between the attacker and the actual point of attack (and there may, depending on the jurisdictions involved, be only limited scope for sanctions if and when an attacker in another country is identified). Moreover, many free-lance hackers publicise their successful methods, via the Internet or otherwise, thus encouraging others to use their methods - and to “improve” upon them. Hence there is an enormous pool of skilled hackers whom the serious attackers can recruit, for fraud, industrial espionage, blackmail, sabotage, etc..

There has been extensive research on formal techniques and on associated tools, such as theorem provers and proof-checkers - and on generalised Boolean methods to make designs more amenable to formal analysis, and formal methods can now make some contribution to confirming the internal consistency of the initial specification and to validating the conformity of hardware or software modules of moderate complexity to that formal specification. However, they can do little to validate the correctness or completeness of the set of requirements embodied in the specification; e.g. no formal method would have detected the “Year-2000 bug”. Validation still depends predominantly on well-designed empirical statistical tests. Furthermore virtually all existing test tools can be fairly readily misused to look for weak spots to attack. Most commercial systems - despite reasonably strict specifications and quite intensive *functional* validation - are far from provably secure. At best they were checked against a list of known threats (but rarely rechecked after subsequent hardware or software changes or against new or newly recognised threats).

Hence any defence must be based on the insertion of a dedicated “trusted” computers and dedicated “trusted” software, at critical points and interfaces. Here the word “trusted” implies that these systems are:

- As limited in functionality, and hence as simple, as possible,
- As rigorously validated and tested as possible,

- Subject to rigorous "cradle-to-grave" hardware and software configuration control,
- Covered by good physical and personnel security,
- Possibly “diversely” duplicated or triplicated (i.e. employing different hardware and different software) where their roles make this appropriate.

Probably few, if any, computers, *as currently procured, installed, maintained, programmed, or operated*, meet these criteria in full but, subject to proper control of the foregoing aspects, it is possible to install and operate an effective trustworthy dedicated interface controller.

Clearly, assurance of the continued integrity of these “trusted” sub-systems is vital. Various technical checks can validate the initial design or detect any subsequent unauthorised hardware or software modifications. All settings, etc., may be subject to dual-key access, but this merely provides traceability, if one of the two then operates independently inside the system.

In many circumstances the continuity of system operation, and the integrity of data, are much more important security considerations than data confidentiality. Modern distributed systems depend on databases for naming, for routing and for holding such things as public key certificates and other more, general directory information. Here there is no requirement for confidentiality (indeed, just the opposite) but any lack of system availability, or any accidental or deliberate data corruption, will have a damaging impact on the operation of the system as a whole. In such situations data replication is a very important and valuable defensive technique.

Since it would be difficult to introduce mutually-consistent unauthorised modifications simultaneously into several distinct systems, the trusted computers might be triplicated, for the most critical applications, with different hardware and different software, so as to give an alarm when one response differs from the other two, and to veto access when all three differ [9]. Simpler variants might use only two systems, looking for agreement, or employ either just diverse duplicate hardware or (more likely) diverse duplicate software. The extra procurement cost of such software replication may be mitigated by the substantially lower standard of design and security validation then acceptable in the constituent systems. However, it should be noted that the implementation of replication can be quite challenging in complex situations, especially where there is a need to allow for non-determinacy (see for example Chapter 6 in [20]).

The system is then split into the smallest acceptable physical and/or functional elements, with all data and control flows to, from or between these elements routed through the trusted controllers, which regulate, *with the tightest operationally tolerable restrictions:*

- Who (user, terminal or device) can “talk” to whom,
- As a member of which defined user community,
- At what security level,
- On what subject category
- In what data format,
- At what peak *and mean* data-rates,
- With what message lengths and message frequencies.

TABLE 6: Traffic-flow controls at a secure gateway

(The commercial “firewall” computers [7], widely used to isolate corporate systems from the Internet, are a particular case of such interface controllers. Although the trust placed in such firewalls may often be excessive, the better ones do perform a valuable function, when properly set up and managed.)

Modular partitioning, with well-defined interfaces, is in fact “best practice" for the design, management and evolutionary upgrading of IT systems. Conversely, however, it can be difficult for interface controllers to function effectively, if introduced retrospectively into ill-structured systems engaged in complex tasks. Even when part of the initial system design, these controllers must maintain a balance between a (probably) vastly improved residual performance in hypothetical hostile environments and a (certainly) somewhat degraded normal performance in the present benign environment.

The controllers also maintain a log of all potentially sensitive traffic, and give an alert for anomalous patterns of transactions. A central organisation must then collate and analyse the global pattern of alerts from groups of controllers, and take prompt action in response - supported by prior contingency planning and by periodic exercises.

Any messages, authenticated by their encryption as coming from a trustworthy source, retain the trustworthiness of that source, irrespective of the message type or of the transmission route. Otherwise, however, even messages or files which do not themselves include any executable commands may be able to trigger such commands or otherwise modify the state of the receiving system. Hence all such traffic can potentially carry communicable “diseases”, such as “viruses”, and one important function of interface controllers is to act as “quarantine” stations preventing such infections spreading beyond sub-system or system boundaries. The growing convergence of computer hardware and software designs onto a small number of industry standards eases the task of these interface controllers but, on the other hand, it increases the impact of any “successful" infection.

In the absence of appropriate cryptographic protection, assumptions as to which data sources are trustworthy and how viral infections may be recognised must be constantly up-dated, in a continuing battle of wits with the designers of viruses. Furthermore, as with conventional quarantine, boundary controls can greatly reduce the risk of infection spreading across a border, but they do not eliminate it entirely, and they can do nothing to control an infection already existing within one of the areas so segregated.

The separation of processing modules by trusted interface controllers may be likened to subdividing a warship by watertight bulkheads: We accept some extra financial cost and impaired freedom of routine communication, as the price of reducing the impact of an attack, from ‘disastrous’ to merely ‘embarrassing’. The project sponsor, leader or user has to judge the right trade-offs between financial, operational, and security desiderata. However, before risking the loss or corruption of an “unimportant” data-source, link or function, he must be sure that it does not indirectly affect a vital operation.

Whilst operational *applications* will generally employ normal commercial hardware and software, some of the most critical ones might use trusted computers - or at least use software and/or hardware diversity, and checking for identical results. The recent trend towards centrally-held applications programs (including Java “applets"), downloaded to simplified user terminals, increases physical vulnerability. However, *provided the transfer of the applications is cryptographically protected*, this may permit improved integrity assurance for these applications programs.

7. Insider Attacks

Within the bounds of a given, commercially- procured system, there can be little defence - other than personnel security - against insider attack at or below this insider's level of access rights. On the other hand, interface controllers can offer significant protection against insider attack spreading beyond the borders of this system segment, and in some system architectures, they could also play a role in preventing an insider going beyond his authorised user group, function or security level.

However, the effectiveness of these encapsulating boundaries depends critically on the interface controllers themselves, and on *their* system managers. Furthermore, the smaller the element segments, the more is not only the aggregate cost of controllers and their impact on a system’s functionality and acceptability, but also the greater is number of system managers involved, and hence the harder is it to maintain a high level of competence and personnel security for these security-system managers. A compromise is inevitable.

The risk of “double-insider” attack by a corrupt *security-system manager* can be very significantly reduced by “double-key” access, making changes in the authorisation lists, authentication code lists and logs dependent on simultaneous joint action by two separate authorised people, with distinct personal authentication “keys”. The price paid for this would be:

- Some reduction in operational flexibility,
- A somewhat increased workload for the security-system managers,
- Possibly some reduction in the morale of the security-system personal.

Protection from network-propagated malicious software is of little value if such software can easily enter the system in other ways. Here the trend towards user-controlled desktop computing resources has complicated the situation since those who have effective ‘day to day’ control over these resources – that is the end users – rarely understand the technicalities of the system. Even systems with sophisticated electronic boundary controls have suffered badly from viruses introduced by their own users uploading software onto their machines.

Education and effective policy measures can reduce this problem, but it takes only a single lapse by a single employee to introduce a virus infection into a large network-connected organisation. This is a particular problem where oppressive software purchasing arrangements engender the temptation to bypass such controls. Worse than this, draconian virus reporting procedures in practice often discourage users from reporting the incidents they experience, thus leaving an infection to spread even when its introduction has been recognised.

Hence, as already indicated, where important centralised information assets are involved, trusted controllers are as relevant at the interface to the internal users as at that to the external world. Such issues have led some organisations to conclude that processing should move back towards centralised computer systems serving multiple “dumb” terminals or “network computers”. Others, however, have moved virus protection from their boundaries onto individual workstations, for the direct protection of the end-users. In practice neither of these models seems likely to dominate future thinking.

8. Physical Attacks on Computer Systems

Some forms of external hacking or insider attacks could render a computer system completely inoperable. Although these may not involve any mechanical or electrical damage to the hardware, their effect is similar, and hence is subsumed by this discussion of defences against physical attacks.

Important computers may have somewhat better physical security protection than many communications facilities. On the other hand, the redundancy in even critical computer systems is normally the minimum for hardware fault tolerance and hence for continuity of operation in the presence of accidental faults. Generally this gives them less resilience to physical attack than is intrinsic in much of the telecommunications infrastructure.

Despite these difference of scale, the basic nature of the threat, and hence the defences, are the same as for the communications network: segmentation to limit the damage when a facility is lost, and redundancy (of data, programs, processing capacity and connectivity) so that the function of the damaged facility can be taken over by an alternative unit with minimal disruption (assuming the applications software is computer independent). Hence computers and their users or user systems should be networked, much like the communications infrastructure, and they need similar network management (software and/or human) to control reconfiguration and possibly adaptation of the *modus operandi*.

Special care has to be taken that any hardware and software involved in providing, and perhaps automating, such recovery and reconfiguration is itself impervious to attack, and to ensure that the various encryption and authentication security features are maintained in reconfiguration, without opening an avenue for attacking or bypassing them.

9. Data Bases

Replicating a data-base n-fold (normally n = 2 or 3) increases its availability by many orders of magnitude. This does however impose special requirements for data-base consistency [12], and careful attention to the replication problems already mentioned in Section 6 above. If data integrity of the database is also ensured by regular cross comparison of the total data content over a suitably secure link, or at least by comparison of parities and “digests”, the database’s integrity will also be dramatically enhanced. On the other hand, the opportunities for - possibly unauthorised - insider access to the data is increased n-fold, and the risk of unauthorised external access may also be increased. The contents of files residing in data-bases, like those in transit, can however be protected by encryption, provided all entitled to access to the information have also access to the relevant decryption key, thus sharply reducing and (hopefully) clearly defining those having potential access to these data .

In most applications, the dramatic improvements in availability and integrity that can be obtained will far outweigh the small increase in the risk to confidentiality, and data replication is then a very important and valuable defensive technique. Indeed, in some applications such as the databases holding public key certificates and naming, routing and other directory information for distributed systems none of this information *need* be kept confidential and indeed some *must* be publicly available, whilst any lack of system availability, or any accidental or deliberate data corruption, would seriously damage the operation of the whole system. (A case in point is the recent major disruption to the Internet caused by accidental corruption of top-level domain naming information [17].) At the other extreme, however, there are a (very) few ultra-sensitive applications, such as a central store of “subscriber-unique” crypto keys, where the data can be replaced by new ones, if necessary, but their secrecy must be absolutely guaranteed, and then replication is clearly not desirable.

Further, more subtle, problem arise when selective user access is provided to databases that contain information at several different levels of confidentiality. For instance, in a database containing confidential salary details, an employee allowed access to his/her own salary information, but also to data summaries and to statistics on salaries across organisational units, may be able to work out the salaries of their colleagues without any direct access to this data. Identifying and limiting the conclusions that can be drawn through such indirect mechanisms is no easy task! For these and other reasons the design of secure databases is a challenging task, but one essential in the evolving world of integrated operation and electronic commerce.

10. Trust Relationships and Security Protocols in Open Systems

In widely distributed information systems - including the global World Wide Web - it is often necessary to establish exclusive relationships of mutual trust between widely dispersed systems elements, thus permitting the dynamic formation of closed domains within an otherwise open system and, in particular, to permit two-way exchanges between unambiguously and undeniably identified users. This capability lies at the heart of electronic commerce, and hence much effort has been expended on the development of appropriate sophisticated cryptographic security protocols.

Clearly, secure data exchange within distributed systems brings major advantages to its users. However, any data exchanges entails some risks and vulnerabilities - some of which are not fully understood, sometimes even by the security-system designers and implementors - thus perhaps lulling the users into a false sense of security.

Cryptographic security protocols permit the secure exchange of data on the basis of a pre-existing relationship of trust. They cannot introduce trust where it does not already exist. For example, in maintaining the confidentiality of data, it is essential to authenticate the identities of those allowed access. This is commonly done using public key cryptography in conjunction with the services of a "Trusted Third Party" (TTP) to ‘certify’ that a particular key is owned by a given, authenticated individual. (In more structured networks, the key-distribution centre performs the same function.) Hence any penetration of the TTP could open wide opportunities for cyber crime, and so technical, physical and personnel security of TTPs is particularly important, and it is essential that the mode of operation of TTPs is well defined and rigorously controlled, and that its relevant aspects are also clearly understood by its users. Duplicating the TTPs at distinct locations, for assured availability, could double the “targets” exposed to penetration for data extraction but, if the two centres have a truly secure way of periodic cross-comparison, for “diversity” operation, then - as in duplicated data bases - it might greatly enhance the assurance of their data integrity.

11. The Human Element

Provided system design makes reasonable technical provision for security, the greatest weakness remaining in the system is that of human failure to:

- Obey the specified operating procedures,
- Observe and report suspicious circumstances, be they:
  - "peculiar" messages or data,
  - unreasonable system operating patterns,
  - suspect human behaviour.

In the absence of a pervasive security culture, at all hierarchical levels, effective defensive precautions may only be introduced or, if present, conscientiously maintained, in the aftermath of a serious attack. An essential element of security is awareness training of all relevant staff, including *update briefings*, when appropriate, *and periodic simulated attacks*. (The traditional reluctance to let simulated degradation of IT capability interfere with routine operations or with training may need to be curbed.) With increasing "out-sourcing" of procurement, installation, maintenance and even operation of IT systems, personnel security, training, briefing and exercising may be as relevant to external contractors as to a company's own personnel. The rotation of contractor staff between *their* employer's various projects, can make the problems of security vetting and security training of that staff more difficult than for intramural personnel. (However, it may be possible to include, in extra-mural contracts, an appropriate security "duty of care", together with stringent penalties for infringements.)

12. Defence in Depth

A major electronic attack on a large distributed communications, computer or data-base, system, e.g. for terrorist purposes, may require careful and detailed covert intelligence collection and analysis and mission planning, just like physical attacks. However, many of the resulting attacks can then be implemented in seconds - if not milliseconds. A defence in depth will therefore comprise:-

- Direct resilience to attack,
- Provision (ideally automated, partially even if not completely) for recovery after attack,
- Resistance to reconnaissance data collection,
- Resistance to effective analysis and interpretation of any data collected,
- Effective maintenance of defence, e.g.
  - applying new security patches to software,
  - monitoring password files for insecure practices,
- Monitoring to detect passive intrusions (or of active interference),
- Noting the methods of attack, and assessing the results obtained by the attacker,
- Collating the results of monitoring from multiple sites and occasions, and analysing them for patterns,
- Trying to identify the attacker and/or his target, and taking appropriate protective or pre-emptive measures.

TABLE 7: Elements of Defence in Depth

This must be backed up by regular "friendly" probing attacks:

- to test the validity and effectiveness of the defence-measures,
- to discover and evaluate any overlooked "holes" in the defence,
- to keep the human element of the defence "on their toes".

Security assurance is not a matter of once-for-all certification, but of continued reassessment, as the system's configuration, its connectivity, its function, its utilisation, its operating procedures - and potential methods of attack - all evolve dynamically. The scale of friendly probing actions must however be limited:- The owners of a system will not wish to disseminate awareness of its residual weaknesses too widely, or to proliferate expertise in attack methods focused onto their system.

13. Dynamic Defence

A further, quite effective and relatively low-cost defence consists of systematic periodic changes in the IT configuration, to render the data collected by an attacker mutually inconsistent - or else consistent but obsolete. Naturally communications and computer systems adapt continuously to changing hardware configurations, a changing user population or organisation, changing service requirements, temporary or permanent changes in traffic-loading patterns, responses to outages or to systems test conditions, updating of routing tables - or of the weighting parameters in routing algorithms. Their explicit security features change as passwords and crypto-keys are updated, key-management systems get up-graded, monitoring facilities evolve, protocols are revised, etc. In the past, in the name of “configuration control”, the aim was usually to keep a network static, even at the expense of technical and operational obsolescence. In the modern, accelerating dynamic environment, systems are (or should be) designed, operated and managed to respond rapidly to technical and operational developments.

If a serious reconnaissance attack on an IT system is detected, it should be possible to take deliberate action to bring about such changes, in the system itself and/or in its security provisions, to render critical parts of the attacker's reconnaissance information invalid and irrelevant.

It is more likely that the attacker succeeds in keeping his reconnaissance covert, in preparation for future passive or active attacks. For large, nation-wide or trans-national systems, such *covert* reconnaissance might occupy many months. Hence, in such systems, an explicit policy of quasi-random changes at a mean interval of, say, 2 or 3 months might ensure that the attacker is for ever chasing his own tail. Since this would largely make a planned virtue of a natural necessity, it should entail negligible cost.

14. Manning Implications of Restoration/Reconstitution

We have already discussed redundancy and reconfiguration as defences against physical - or equivalent software - attacks. Quite powerful software tools are available for automatic adaptation to “normal" changes of availability and capacity. However, serious damage will demand a sophisticated combination of the recognition of patterns, assessment of options, weighing up of the balance of conflicting factors, and recognition of relative importance in a given operational scenario, all of which need human judgement. The system-monitoring facilities must however provide a user-friendly interface permitting these judgements to be effectively exercised.

This applies to deciding and implementing the appropriate reconstitution action and to the initial management of the resulting, reconfigured system This activity can probably be restricted to a short peak of high-intensity, high-adrenaline effort (say between 2 and 48 hours). Since manning one post for 24 hours/day, 365 days/year requires in practice a total of 5 people, it should be possible to double the staff during such a brief period without any increase in complement. However, provision for fall-back to less automated operating modes may limit the options for manpower savings. Any "reserve" staff, to deal with contingencies, must have a meaningful regular task, which maintains the skills they will need in an emergency.

15. Integrated System Management

Information-system management deals with aspects of connectivity, capacity, load-factor, quality-of-service, reconfiguration, prioritisation, etc. It is also responsible for making the best of a system damaged by attack. The management of interface controllers has to balance some of these considerations against sometimes conflicting security requirements. Monitoring the pattern of attack attempts, assessing their significance and co-ordinating any response, has many similarities to conventional system management. However, it must also use any clues to the methods of the attacker, his objectives, his capability, the amount of information he may have gleaned, and possibly to his identity. Such information-security management is best handled as an explicit, important and substantial enlargement of conventional information-system management. Systems managers should welcome this enrichment of their function, with all its operational importance and professional challenge. This is a very effective action that can be implemented almost immediately.

16. Prioritisation in the Selection of Defence Options

There is no complete defence against cyber crime, certainly none that is affordable! We have identified three basic components of fairly universal validity:-

1.	Security awareness training.
2.	Integrated security and system management
3.	System redundancy

TABLE 8: Basic Defence Components

However, there is a further set of defences whose scope nature, scale, and areas of application, in a resource-limited situation, have to be guided by prioritisation:

4. Bulk encryption of important trunks links
5. End-to-end encryption of important permanent or temporary connections.
6. Appropriate crypto key-distribution systems.
7. Incorporation of authentication and similar features in the crypto schema.
8. The extensive use of trusted secure interface controllers.
9. Dual or triple diversity in critical processors and/or software.

TABLE 9: Higher-Level Defence Components

To achieve this prioritisation, we must:-

1. Relate our IT components to the functions served
2. Identify those IT threats which - if not countered - are sufficiently likely to arise and sufficiently disastrous operationally that they *must* be mitigated
3. Identify those defensive measures which:-
    - are most effective against these threats
    - are most affordable
    - least impair the system's normal operation
    - are least system- or scenario-specific
    - give most collateral benefit the defence of other (lower priority) systems or functions
    - give most collateral benefit to the defence of the high-priority systems or functions against other (lower priority) threats.

TABLE 10: Principles of Prioritisation

It is then a matter of judgement how far down the resultant list of threats and countermeasures to go. With given resource constraints, we might have to accept, say, a remote risk of loss of confidentiality of information to counter the risk of a disastrous loss of operational capability. Since both the technology and the threat are steadily evolving, the overall defence policy and its detailed implementation (such as the settings for specific interface controllers) must be kept under continuing or regular review.

The potential targets to be protected are not just the IT system(s) of the relevant company or organisation, but:

- All entities within this company or organisation dependent on its IT system(s).
- All entities up-stream from that organisation which provide its input.
- All entities down-stream which accept (and depend on) its output.
- Any entities which rely on two-way real-time interaction with that organisation.

(Since virtually everyone is dependent on telecommunications, energy, water and transport utilities, and financial services, it is of special importance that all those local or national, general or dedicated IT systems which are critical to these services are sufficiently resilient.)

Relative defence priority is essentially the product of impact and probability. The *impact* of the threat, if it should materialise, is scaled down by any compartmentation, which restricts the scope of the damage (but increased by standardisation within the compartments), and it may also be reduced by any provision for rapid recovery, provided such recovery *significantly* lessens the operational effect. The *probability* of the threat is essentially *the attacker's* priority. This is determined mainly by the “attractiveness” of the target - from his point of view - modified (positively) by his perception of probability of success and (negatively) by his assessment of likely resource cost to him, and by the likelihood of his suffering effective retribution. (Even apart from the above modifying factors, the attacker's priority list is unlikely to be identical with the defender's, because of their distinct constraints and objectives.)

The suggested combination of impact and probability focuses the defensive effort onto those systems which come high on *both* our own impact priority list and our assessment of the potential attacker's priority list. The defence should then aim to reduce all these threats to a similar, acceptably-low probability of success, or else to reduce the impact of any technically successful attack to an acceptably low level.

17. Conclusions

The growing dependence of industry and society on IT, and the growing threat of cyber crime, require that serious effort be devoted to IT security. Due to interdependence and interconnection beyond any one system's or organisation's boundaries and responsibilities, the usual parochial and short-term focus of IT security is no longer adequate. We must consider the possible impacts on or from, other systems (often in other organisations), whether caused by accidental hardware, software or operational faults, or - as analysed in this paper - by malicious acts, including possibly by insiders.

By way of conclusions it is appropriate to reiterate the following points:

1. Just as even the best safety-critical systems fail at times, due to an unforeseen fault or - more typically - combinations of faults, so protection against skilled determined attackers is unlikely to guarantee complete success. Nevertheless, much can be done to reduce the danger of such threats.
2. Everyone concerned with the design, acquisition, operation, management or use of any IT system, on which people or organisations are highly dependent, must be aware of the relevant security issues and of his own security responsibilities.
3. A realistic security policy must seek to, and in many cases can, limit the impact of any reasonably likely attack to a (just) tolerable degradation or interruption of service
4. This degree of immunity can in many cases be achieved at a tolerable marginal cost in system procurement and operation, coupled with a tolerable degradation of the quality of “normal” service to the user.

5. This does not merely entail technical measures, such as fault tolerance, encryption, compartmentation, etc., but it also requires continued close supervision by well-informed management.
6. One important strategy, especially when using limited-security systems and sub-systems, involves placement of (justifiably) trusted interface controllers to monitor and restrict information flow at appropriate points, both between sub-systems and between a system and the outside world - including its own users. (Internet “firewalls” are one attempt at such interface control.)
7. Such interface controllers necessarily impose limitations on the facilities provided to users, and on the forms of system interconnection permitted.
8. The security measures must be- and must be seen to be - both technically adequate, and operationally acceptable. Only then will they be loyally adhered to by all involved.
9. Without careful attention to these issues, the uncontrolled interconnection of existing systems, many on which people and organisations are critically dependent, will continue to create huge, ill-defined and defenceless super -systems.
10. **The price of protection against from** *cyber crime* **is eternal** *IT security* **vigilance.**

References

1. ANDERSON, R.: ‘Why Cryptosystems Fail,’ *Communications of the ACM*, 1994, **37** (11), pp.32-40.
2. ANDERSON, R. and KUHN, M.G. ‘Low Cost Attacks on Tamper Resistant Devices’, Security Protocols - 5th International Workshop (LNCS vol. 1361), 1996, Springer Verlag, pp. 125-136.
3. ANDERSON, R. and KUHN, M.G. ‘Tamper Resistance - a Cautionary Note’, Proc. Second Usenix Workshop on Electronic Commerce, 1996, pp. 1-11.
4. APT, K.R. and OLDEROG, E.-R.: ‘Verification of Sequential and Concurrent Programs (2nd Ed.)’ (Springer-Verlag, 1997).
5. BENJAMIN, R.: ‘Security Considerations in Communications Systems and Networks,’ *Proc. IEE*, 1990, **137, Pt 1** (2).
6. BLAIN, L. and DESWARTE, Y. ‘Intrusion-tolerant security servers for Delta-4’, Proceedings of the ESPRIT’90 Conference, 1990, Brussels, Kluver Academic Publishers, pp. 355-370.
7. CHESWICK, W. and BELLOVIN, S.: ‘Firewalls and Internet Security: Repelling the Wily Hacker’ (Addison-Wesley, Reading, Mass., 1994).
8. DESWARTE, Y., BLAIN, L. and FABRE, J.-C. ‘Intrusion Tolerance in Distributed Computing Systems’, Proc. 1991 Symp. on Research in Security and Privacy, 1991, Oakland, California, IEEE Computer Society Press, pp. 110-121.
9. DOBSON, J.E. and RANDELL, B. ‘Building Reliable Secure Systems out of Unreliable Insecure Components’, Proc. Conf. on Security and Privacy, 1986, Oakland, IEEE.

10. GENERAL ACCOUNTING OFFICE (GAO). 'Information Security - Computer Attacks at Department of Defense Pose Increasing Risks', GAO/AIMD-96-84, U.S. Senate, May 1996.
11. GHOSH, A., DEVADAS, S. and NEWTON, A.R.: 'Sequential logic testing and verification' (Kluwer AP, Boston,1992).
12. GRAY, J. and REUTER, A.: 'Transaction Processing: Concepts and techniques' (Morgan Kaufmann, 1993).
13. GUPTA, A.: 'Formal hardware verification methods: A survey,' *Formal Methods in System Design (Kluwer AP)*, 1992, **1** (2/3), pp.151-238.
14. HOBBY, J.: 'Internet Crime: Cyber Leeches,' *Computer Weekly*, May 12, 1996.
15. KLUEPFEL, H. 'Countering Non-Lethal Information Warfare', Proceedings of the IEEE 29th Annual International Carnahan Conference on Security Technology, (1995).
16. LAPRIE, J.C. 'Dependability: From concepts to limits (Invited paper)', Proc.12th Int. Conf. on Computer Safety, Reliability and Security (SAFECOMP'93), 1993, Poznan-Kiekrz, Poland, Berlin: Springer-Verlag, pp. 157-168.
17. MARKOFF, J.: 'Ignored Warning Leads to Chaos on the Internet,' *New York Times*, 18 July 1997.
18. NEUMANN, P.: 'Computer Related Risks' (Addison-Wesley, New York,1995).
19. PERRY, W.: 'Effective Methods for Software Testing' (John Wiley & Sons, Inc., New York,1995).
20. POWELL, D. 'Delta-4: A generic architecture for dependable distributed computing', Research Reports ESPRIT (Vol. 1), (Springer-Verlag, Berlin, Germany, 1991) pp.484.
21. PRESIDENTIAL COMMISSION ON CRITICAL INFRASTRUCTURES PROTECTION: 'Commission Report', Washington DC, 1997.
(http://www.pccip.gov/report_index.html)
22. REITER, M., FRANKLIN, M., LACY, J. and WRIGHT, R.: 'The Omega Key Management Service,' *Journal of Computer Security*, 1996, **4**, pp.267-287.
23. SCHWARTAU, W.: 'Information Warfare: Chaos on the electronic superhighway' (Thunder's Mouth Press, New York,1994).
24. WARD, M. 'Supplementary Paper', European Conference on Security and Detection, (April 1997).

The Authors:

Ralph Benjamin, CB, PhD, DSc, FIEE, FCGI, F.Eng.
13 Bellhouse Walk, Kingsweston, Bristol BS11 OUE
tel/fax: +44 117 982 1333, E-mail: DrBenjamin@aol.com

Brief CV:
Joined Royal Naval Scientific Service in 1944. Following consecutive individual-merit promotions, Deputy Chief Scientist and Head of Research, Admiralty Surface-Weapons Establishment, Director and Chief Scientist, Admiralty Underwater-Weapons Establishment, Superintendent Director and Chief Scientist, GCHQ. Following 5 "post-retirement" years as Head of Communications Techniques and Networks, Nato Supreme HQ, has, since 1988, been a consultant, and Visiting Professor to a number of universities - currently Imperial College and University College, London, and University of Bristol.. IEE Marconi Premium and Heinrich Hertz Premiums (twice). Sole or principal author of over 100 Learned-Society articles and two books: Modulation, Resolution and Signal Processing (Pergamon, 1966) and Five Lives in One (Parapress, 1996).

Brian Gladman, Ph.D
4 Palmers Green, St Johns, Worcester WR2 4JY
E-mail: gladman@seven77.demon.co.uk

Brief CV:
Joined the Royal Naval Scientific Sevice in 1964 and undertook research on phased array antenna systems. After a period working in the United States he returned to work on computer automation for ship-based command and control systems. Moved in 1980 to lead computer applications research and development at the Royal Signals and Radar Establishment, Malvern where he directed research on formal methods for software design and development. After retirng from the Ministry of Defence in 1994 he became Deputy Director of the NATO SHAPE Technical Centre. He is now an independent consultant specialising in Information Security.

Brian Randell, ARCS, DSc, FBCS, C.Eng.
Department of Computing Science, University of Newcastle, Newcastle upon Tyne, NE1 7RU, UK
tel: +44 191 222 7923fax: +44 191 222 8232, ,
E-mail: Brian.Randell@newcastle.ac.uk, URL: http://www.cs.ncl.ac.uk/~brian.randell/

Brief CV:
Joined the English Electric Company where he led a team which implemented a number of compilers, including the Whetstone KDF9 Algol compiler. From 1964 to 1969 he was with IBM, mainly at the IBM T.J. Watson Research Center in the United States,

working on operating systems, the design of ultra-high speed computers and system design methodology. He then became Professor of Computing Science at the University of Newcastle upon Tyne, where in 1971 he initiated a programme which now encompasses several major research projects concerned with reliability, security, and distributed systems. He is Project Director of DeVa, the ESPRIT Long Term Research Project on Design for Validation and of CaberNet, the ESPRIT Network of Excellence on Distributed Computing Systems Architectures, and has published nearly two hundred technical papers and reports, and is co-author or editor of seven books.