Observations on information security crisis

Jussipekka Leiwo
jussi@kmitnb.ac.th
King Mongkut's Institute of Technology North Bangkok, Faculty of Applied Sc.
1518 Pibulsongkram Rd., Bangkok 10800, Thailand

Abstract

Despite a wide body of academic knowledge of secure information systems, application software, communication protocols and cryptographic primitives remain insecure. This is especially alarming in the emerge of application domains and organisational structures that depend heavily on the availability of reliable and secure data communication infrastructure, such as electronic commerce. A survey of recently reported vulnerabilities demonstrates that systems remain susceptible to attacks known for decades. The lack of security awareness among system and protocol designers and therefore occurring security problems are called the information security crisis. This paper surveys the symptoms and causes of information security crisis, and sketches an outline of an approach required for tackling the crisis.

Keywords: Data security, Computer communication systems

BRT Keywords: USE, UF

Introduction

Since the information theory based cryptography by Shannon (1949) and the public key cryptography (Diffie & Hellman 1976), cryptology has become a major research area. In addition to obvious applications in encrypted and authenticated communication, cryptographic protocols have been devised for, for example, electronic payments and voting, mental poker and fair coin flipping (*e.g.* Schneier 1996). In parallel to cryptology, research in multilevel secure (MLS) computer systems, based on models by Bell and LaPadula (1973) and Denning (1976), has significantly advanced the knowledge of secure computer systems and databases. MLS systems require dedicated hardware and specialised operating systems, not available in commercial systems typically connected to open, public networks, such as the Internet. Therefore, their applicability in the security of commercial systems is limited. High cost of MLS systems further reduces the applicability. Since MLS systems also do not adapt well to networked environments, the security of electronic commerce, for example, is mostly concerned with the security of communications, usually achieved by standardised security features, for example through Secure Socket Layer (SSL) protocol underneath the Hypertext Transfer Protocol (HTTP). MLS systems are only found in central key servers and key certificate databases. Denning (1999, p.377) estimates the cost of a typical minimum level (C2) certification in TCSEC evaluation criteria ranging from US$500,000 to US$800,000. She also reports cancellation of a project to develop an operating system at the highest security level (A1) after years of development due to a low expected volume of sales.

It is possible, at least in theory, to develop secure systems. However, information

systems remain insecure. Operating systems, application software, communication protocols, cryptographic primitives and, as a result, entire systems remain susceptible to simple attacks known for decades. An analogy is drawn to the software crisis that was concerned with difficulties of software products meeting client requirements and schedules at the early years of the large scale system engineering (Pressman 1997). The term 'information security crisis' is used to describe the dilemma of a wide body of scientific knowledge of information security not translating into an improved security of information systems. To demonstrate the severity of the information security crisis, common security violations are studied and classified into software security and communication protocol security problems. Since some recent attacks demonstrate problems with the actual cryptographic algorithms, a brief note shall be provided on the problems with security primitives. More comprehensive classifications of vulnerabilities are available (*e.g.* Denning 1999, Escamilla 1998, Neumann 1995). Rather than aiming at the same depth, well-known attacks are summarised to demonstrate the vulnerabilities.

Systems remaining vulnerable to attacks known for long periods of time leads to two conclusions. First, new approaches are required towards the security of information systems. If a substantial body of scientific knowledge can not be applied in the development of information systems, it is essential to question the methods used for developing secure system from presumable secure components, such as cryptographic primitives and protocols. Second, since many vulnerabilities originate from the design of communication protocols and programming languages, the entire paradigm for information system development should be reconsidered with security being an integral design criterion. Escamilla (1998) uses similar arguments to justify the need for the detection of security violations. Security technologies aim at preventing security violations. Since the main reasons for security violations are configuration errors and software errors, there is also a need for intrusion detection to strengthen the security. Intrusion detection is a valuable tool for security administration, as are firewalls and vulnerability scanners. However, intrusion detection systems are complex software products especially hard to configure. They remain potential targets for attacks and must be implemented using the same vulnerable system development tools than other software. Therefore, more radical approaches are required for information security.

The problem of poor security especially emerges in the context of electronic commerce over open, public networks. Information security is essential in electronic commerce (Borenstein *et al.* 1996). Yet, common mechanisms for interconnecting systems are highly insecure. Even though the application level protocol can be secured, the vulnerabilities of underlying protocol structures may lead to security problems. In the emerge of electronic commerce and other applications that heavily on the communication infrastructure, it becomes fundamental to understand security vulnerabilities of the past, and device means for encountering these vulnerabilities in the future.

This paper starts by identifying symptoms of information security crisis, followed by a study of the causes of security problems in information systems. The two emerging strategies, establishment of the scientific foundation of information systems security and the development of less prohibitive approaches towards information systems security, are then identified as potential means for tackling the information security crisis. Finally concluding remarks shall be provided.

Symptoms of the crisis

Problems with information systems security are classified into software security

problems, communication protocol security problems, and problems with cryptographic primitives. Software security problems are a fundamental class, since the failures of adequately implementing security enforcement technologies and ignorance to widely known security vulnerabilities lead to common system break-ins. Some major software vulnerabilities shall be summarised to demonstrate the lack of security awareness in software design and implementation. An especially concerning characteristics of these attacks is that they employ techniques widely known for more than a decade, hence indicating poor interest in security in software engineering.

It is widely known, that application of secure cryptographic primitives does not necessarily lead to secure communication protocols. Sequence and structure of protocol messages may lead to the disclosure of data or other types of attacks. Not only cryptographic protocols, but also many commonly used internetworking protocols are vulnerable. Common vulnerabilities in widely used communication protocols, mostly related to the TCP/IP suite, shall be discussed next to demonstrate the lack of security awareness in protocol design. Finally, security problems with actual security primitives shall be discussed briefly. This discussion demonstrates that difficulties in proving cryptosystems secure may lead to wide applications and standardisation of primitives later proven insecure.

Software security problems

Early examples of attacking hosts connected to the Internet through software vulnerabilities include those exploited by the so called Internet Worm (Spafford 1989). Among guessing passwords and exploiting poor security administration, the Internet Worm exploited weaknesses in application design, causing buffer overflows in the program execution stack. Since the program execution stack is usually executable memory, the overflow can be engineered to cause execution of arbitrary system commands in the receiving host, usually operating system shells on root privileges. Even though these problems have been well known since the Internet Worm, many applications are still susceptible to similar vulnerabilities. For example, the MIME-Bug (CA-98.10[1]), exploits the same vulnerability. Computer Emergency Response Team (CERT) archives also indicate that most known attacks on systems still exploit potential for buffer overflows either to gain privileged access or to cause a denial of service. Table 1 classifies security incidents reported by CERT in 1998 and 1999 into buffer overflow attacks, Denial-of-service attacks, trojan horses and other attacks. Asterisk indicates that an incident falls into several categories. All buffer overflow attacks use the same mechanism, known since the Internet Worm, to gain unauthorised access. Some attacks also combine denial of service with buffer overflow. In fact, the number of software vulnerabilities is so high that practical guides for UNIX security suggest all daemons being reprogrammed as part of the security project (Cheswick & Bellovin 1994).

Despite poorly implemented software, various forms of malicious software may cause severe problems due to improperly implemented operating system memory protection. Despite traditional viruses and other type of malicious software, new types of viruses have been implemented to exploit weaknesses in various application software packages and macro languages. For example, the recent MS-Word Macro Virus Melissa (CA-99.04) appeared extreme severe. Some CERT reported incidents are classified as

[1] All CERT advisories are indicated by the official CERT identification. For example, CERT Advisory CA98-10 is the 10th published advisory at 1998. All advisories are available at http://www.cert.org

Table 1: CERT reported security incidents 1998-1999

Buffer Overflow	Trojan Horse	Denial of Service	Other
CA99-04	CA99-02	CA98-13	CA98-07
CA99-03	CA99-01*	CA98-05*	CA98-03
CA99-01*		CA98-02*	
CA98-12		CA98-01	
CA98-11			
CA98-10			
CA98-09			
CA98-08			
CA98-06			
CA98-05*			
CA98-04			
CA98-02*			

Trojan Horse attacks. The characteristic is, that malicious software that exploits a vulnerability is included in the system, and later used for attacking the system. These attacks do not directly result from poor security awareness of application software engineering. Rather, poor operating system design is exploited to either hide malicious software or overwrite sensitive data.

Industry standards have been specified for application programming interfaces for security services, such as GSS-API (Linn 1997). Since software security problems, such as buffer overflows, are caused by vulnerabilities in standard function libraries, they can not be prevented by extensions into languages. Even if the buffer overflow problems can be easily avoided, they appear commonly. Buffer overflow is characteristics to programs implemented in C language. Languages such as C++ and Java are not as vulnerable.

Pfleeger (1997, Ch.) lists two major reasons for the existence of security problems in software. First, controls of programs apply at individual program or programmer level. This implies difficulty of detecting well-hidden malicious code in a software artefact. The complexity of software products prevents extensive evaluation of source code. Second, software engineering technologies evolve far more rapidly than security enforcement technologies. As a consequence, security research has difficulties applying existing security technologies to new software engineering technologies. The first reason is clearly administrative and requires thorough software engineering process to be tackled. The second is an implication of the difficulty of integrating existing security technologies in system engineering models.

Controls listed by Pfleeger are well-known quality controls for system engineering, such as peer reviews. Software process improvement paradigms can be extended to cover security of software. The US Department of Defence Standard 2167A (1998) for software engineering covers the whole system development life cycle. The NSA adaptation of the Capability Maturity Model, SSE CMM (1995), establishes a similar approach to the security process improvement than the original CMM to the software process improvement. Since software security flaws appear at each stage in the system development life cycle (Landwehr *et al.* 1994), process improvement is a significant security improvement paradigm. The problem with process improvement, however, is that it is an administrative security measure, hence not being capable of fully solving the security problems caused by programming languages or operating systems.

Network protocol security problems

Despite through vulnerable software, systems can be attacked through vulnerabilities of data communication protocols. Several well-known examples of vulnerabilities have been published, for example the TCP SYN flooding attack (CA-96.21), the TCP/IP Ping

attack (CA-96.26) and Microsoft PPTP vulnerabilities (Schneier & Mudge 1998). TCP SYN flooding and Ping attacks are examples of emerging denial of service attacks. TCP SYN flooding attack exploits a vulnerability in the TCP session establishment procedure of the TCP/IP protocol suite. A large number of session requests are initiated but not completed. As initiated session requests are stored in a finite sized buffer in the target host, eventually the buffer will overflow and further session requests from both legitimate and illegitimate sources are denied. Ping attack exploits a vulnerability in common implementations of the Ping protocol. Ping uses ICMP (Internet Control Message Protocol) of the TCP/IP suite to query the status of other hosts. The attacker sends oversized ICMP messages to the target host and causes an overflow in the receiving data structure. As Ping-daemon operates on the root privilege in most systems, the overflow can cause overwriting of any memory segment in the target host. Large number of attacks will eventually result in system failures due to the overwriting of system critical data.

Vulnerabilities of the Microsoft PPTP protocol can be exploited to virtually any type of an attack and may result in a total loss of security, mostly due to inappropriate protocol design. For example, after the Internet Worm incident, the UNIX password encryption mechanism was modified to include so called salt value that prevents off-line exhaustive search of all possible passwords. Previously, it was possible to steal a password file (world readable file /etc/passwd in early UNIX systems) and mount an off-line attack where a large number of password candidates are encrypted and compared to the instances in the stolen password file. Microsoft PPTP protocol does not use salt in password encryption, and some versions convert all passwords to uppercase before encrypting them, thus significantly reducing the number of password candidates. Design decisions required to maintain backwards compatibility make some attacks even easier.

Literature in cryptology highlights the difficulty of constructing secure protocols from presumably secure cryptographic primitives. The problem of composing secure protocols can be further highlighted by results of applying various logics for analysing security protocols. Introduction of, for example, BAN logic (Burrows *et al.* 1990) for evaluation of security protocols has helped to prove insecure several protocols assumed secure. These results are not only of academic interests, their pragmatic value lies in the assistance to selection of communication protocols for being used when constructing systems with security requirements. As protocols can be proven to be secure, demonstration of security should be a basic criteria for new protocols.

In networked operating systems, the security of systems depends on communications security, operating system security and security of application software. Some security functionality can be achieved through dedicated hardware, such as cryptographic modules implemented on tamper-proof hardware. Communications security is difficult to achieve without trusted third party services, such as public key authentication infrastructures. Many security architectures focus especially on the inter-process communication and prevent bypassing of security controls implemented as part of application software. As a consequence, frameworks for the development of secure systems and applications, such as Kerberos (Steiner *et al.* 1988) and Sesame (Kaijser 1998), focus on a comprehensive security architecture instead of only securing communication protocols or actual application development tools. Standard network services can then be re-engineered to employ the underlying security architecture.

Anderson (1993) concludes that most attacks originate either from human behaviour or improper implementation of security models. Human behaviour is a complicated issue, and it is generally acknowledged that most computer crime is committed by insiders of organisations. It is difficult to specify technical security

measures that can prevent misuse of information by those authorised to access it. Rather, various operational and administrative measures are required to deal with human issues in information security. Appropriate implementation of security models, on the other hand, is the primary goal of various security evaluation criteria, most importantly emerging Common Criteria for IT Security Evaluation (1998), discussed later.

Security primitive problems

A recent discovery by Bleichenbacher (1998) of a security flaw in the public key cryptosystem standard PKCS#1 demonstrates that vulnerabilities do not always occur through presumably weak links; operating systems, communication protocols and application software. The attack exploits vulnerabilities in a presumably secure protocol and underlying encryption algorithm. Bleichenbacher devices a chosen ciphertext attack against RSA cryptosystem, and even thought the number of chosen ciphertexts is considerably large, between 300,000 and 2,000,000 chosen ciphertexts decrypted by a specific server, it is still claimed to be feasible (Shoup 1998). What makes the attack especially significant is that the Secure Socket Layer (SSL) protocol, widely used for protecting World Wide Web traffic, also uses that standard. The importance of the attack is that it underlies the assumption of security algorithms and protocols being the strong link in systems, therefore highlighting the difficulty of design and implementation of good security enforcement.

Since evidence of security of various cryptographic primitives is usually indirect, new discoveries in cryptanalysis often change the conception of security of various security primitives. For example, introduction of differential cryptanalysis (Murphy 1990) and linear cryptanalysis (Matsui & Yamagishi 1992) have proven weak presumably strong cryptosystems and hash functions. Yet, well designed cryptosystems, such as DES (1977), remain immune on these attacks[2]. The knowledge of new cryptanalytic techniques is, of course, subsequently translated into the knowledge of proper cipher design but developments in cryptanalysis may prove systems assumed secure insecure even decades after publication and wide applications. In fact, many cryptographers go into considerably length explaining why only cryptosystems able to withstand extensive public scientific scrutiny should be used in any system.

Causes of the symptoms

A number of reasons shall be identified for causing failures in information systems security. The list is an extension of the summary published in (Leiwo *et al.* 1999).

Lack of mechanisms for evaluating security

Advances in computer security models led to the establishment of several security evaluation criteria. The core of security evaluation is testing the design and

[2] The participation of NSA in the design of DES and classification of fundamental design principles raised concern about the possibility of intentional weaknesses in DES design to enable easy governmental disclosure of encrypted data. Yet, DES has proven secure against attack techniques deviced decades later. Currently, DES has been proven inadequate because of the 56 bit key length. A contest has been deviced for the new Advanced Encryption Standard (AES) for the protection of unclassified communication and candidate algorithms are currently under evaluation.

implementation of the trusted computing based (TCB), the security enforcement functionality of a trusted computer system, to establish a specific level of assurance of correctness. The core idea originates from the MLS research and provides little flexibility for evaluating comprehensive systems. The Common Criteria (1998) appears promising alternative for system evaluation but has just recently been standardised. Therefore, the acceptance rate is not yet known. Alternative mechanisms for assurance include the ISO 9000 approach (vonSolms & Meyer 1995), extensions to CMM (SSE-CMM 1995), extensive checklists (Kwok & Longley 1997), and self audit procedures (vonSolms 1996). Security evaluation is an emerging area of research. However, as shall be discussed later, the unsuitability of positivist case studies in information systems security makes it difficult to compare different evaluation strategies.

A gap between management and enforcement of information security.

Security models do not support integration of security design into overall system design. This development duality of information systems and security leads to severe information security difficulties (Baskerville 1992). Requirement engineering and object oriented modelling techniques have been successfully used in the specification of computer security requirements (Boswell 1995) but little support is provided for the comprehensive security of information systems. Several layers of elaboration are required in the development of secure systems and security technologies are of concern only on lowest layers (Williams & Abrams 1995). Management of information security should clearly support tasks at all layers and the focus of research should be on these tools. Also, there should be a seamless refinement path of high levels of abstraction into technical specifications of security enforcement measures. Some models have been proposed for formal derivation of technical security requirements from abstract security policy objectives (Leiwo *et al.* 1999) but most models focus on security requirements based on risk analysis instead of dealing with abstraction of security requirements independently from the cost of implementation of those mechanisms.

Conflicts with top-down system design principles.

As system design methods, information security design methods should be top-down. Early phases of system design aim at specifying an abstraction of a system to be implemented. System architecture, implementation tools and methods, and underlying enabling technologies are not known. Yet, most models for the management of information security depend heavily on risk analysis. The core of risk analysis is identification of various vulnerabilities and threats before protection measures can be specified. The problem is that vulnerabilities originate from underlying implementation technologies that are not known at the early stages of security design. Therefore, new approaches should enable processing of incomplete abstractions of information security requirements and refinement these primitives once decisions regarding implementation technologies are made. One such approach is harmonisation of information security requirements (Leiwo *et al.* 1999) that attempts on establishing flexible coordination of existing security enforcement technologies to achieve flexibility in the design of information security safeguards.

Despite being subjected to severe critics since the introduction to the security of information systems, risk analysis is widely seen as the main tool in the design of information security. Survey of critics by Backhouse and Dhillon (1996) focuses mostly on the limitations of the theory of probability in estimating losses and cost of protection. VonSolms (1996) points out high cost and low speed of risk analysis and dependency on subjective decisions. Baskerville (1991,1993b,1995) identified the weakness of risk analysis as a scientific method, the primitiveness of risk analysis as a modelling

technique, and the inadequacy of risk analysis dealing with business issues of information security. Because of the cost reducing, not profit-generating, nature of information security, business decisions must be made about desired protection, it is difficult to find alternative mechanisms for risk analysis. For example, (Leiwo *et al.* 1999) attempts to reduce the role of rather than replace risk analysis.

Lack of support for information security in non-traditional organisations.

Inter-organisational networking has enabled dynamic organisational structures, where a key success factor is security of information and rapid adaptation into changing operational environments (Borenstein *et al.* 1996). Mechanism for information security development should also enable modelling of organisation according to responsibilities and authorities concerning security, not the business structure. Backhouse and Dhillon (1996) deal with structures of responsibility related to information security, and Leiwo *et al.* (1999) establish a model for formally modelling an organisation for information security development independently from the business structures. Flexibility appears a key success factor of information security, especially in adaptive organisations, making it hard for traditional security development methods being applicable in new types of organisations (Baskerville 1991,1993b,1994,1995).

Lack of consensus of definitions of concepts involved.

Due to the wide scope, there are doubts whether concept "information security" is clearly understood by researchers and practitioners. Large number of conflicting definitions make it difficult to establish a commonly accepted framework for information security. The fundamental success factor of any security model is its capacity of catching the nature and formulating intuitive concepts, such as "information security" (Bell 1988). Because of the importance of non-technical considerations in the security of information systems, definitions of these concepts must be kept wide. Models for comprehensive security of information systems range up to legal and ethical considerations (Kowalski 1990), social and ecological layers (Hartmann 1995) and human group behaviour (Leiwo & Heikkuri 1998). These models are valuable tools for studying socio-ethical impact of information systems security but contribute little to the actual development of secure systems. From the modelling perspective, high level issues should not be the target of modelling but models should enforce them as constraints to the modelling. Yet, a common research framework for information systems security should adopt many different views towards information security.

Scientific difficulties in information systems security research.

Information systems security is a considerably new scientific discipline and is still on a pre-methodical phase. Baskerville (1994) identifies a number of areas of future work in information systems security, some of which focus on the establishment of a scientific foundation for further research. Security vulnerabilities are not usually publicly known, and information security related information is usually sensitive. As a consequence, the positivistic and post-positivistic research based on empirical case studies widely applied in information systems research may not be successful. Additional complication is caused by the difficulty of detecting successful security violations, implying inappropriateness of quantitative methods. Blakley (1996) claims that 88% of penetration attacks involving real systems were successful. Of all attacks, 96% went undetected and in 95% of detected cases, no follow-up action was taken. Denning (1999, p.373) reports similar findings from other experiments.

In fact, the entire question of research models and research methods in information systems security remains mostly unanswered. A significant area of further

research is required to establish the scientific domain ‘information systems security’ and to study its relationship to the information systems domain. Obviously, information security is attempting to become an independent area of research. A number of conferences and journals are dedicated in information security, but the focus is often unclear and topics range from computer security models and cryptographic primitives to socio-ethical and legal considerations. Most results published focus on a specific technical problem and alternative solutions available for that particular problem with little generalisation being possible of findings. Studies on research models or methods are rare. Due to an increasing business interest in secure information systems and general increase in security awareness, it is of utmost importance questions of the scientific foundation of information systems security to be studied in detail.

How to tackle the causes?

Surprisingly, software crisis has not been subjected to a large number of academic research. Research in computer aided software engineering (CASE) and relational databases frequently justify the research by software crisis but the concept itself is not subjected to much detailed research. Fitzgerald (1990) makes an observation, that most methodologies for the development of information systems concentrate on the situation given at design time and allow only little flexibility on future changes. Boogaard (1994) extends this observation to conduct research on data independence as a tool for achieving flexibility, claiming that flexibility of design methods is the major tool for defusing the software crisis.

Flexibility of security safeguards is also one of the major still unanswered research questions in information systems security. Flexibility should be integrated in both security measures and mechanisms for specifying these measures. As pointed out by Fried (1994), information technology field changes rapidly and many new technologies alter the threat scenarios significantly. Baskerville (1995) differentiates between first-order and second-order issues in security. First-order security is concerned with the technical foundation for security, and security mechanisms justified by first-order arguments, such as traditional risk analysis, establish organisational power structures that prevent flexibility in operations disabling rapid adaptation into unexceptional situations. This easily leads security becoming a preventive instead of enabling factor, preventing adaptation into changing operational requirements. The limited scope of first-order considerations should be extended to organisational considerations, such as the business risks the organisation is facing and whether that risks justifies the cost needed for protection.

Comprehensive security of information systems requires contributions from many scientific fields, at least theory of computability to justify and evaluate security measures, computer and communications security to establish a model of security, software engineering to adequately implement the security model, system analysis and design to capture the nature of security requirements, and socio-ethical considerations to establish and enforce operational procedures and guide lines for information security. To establish a scientific foundation for information systems security, existing frameworks from related disciplines need to be considered from the security point of view. Despite some fundamental differences, the field of dependable computing has been applied in security context (*e.g.* Jonsson 1998). The core is to establish an interpretation of security and dependability that allows incorporation of them into a unified, quantifiable framework. By dividing security into behavioural and preventive measures, some facets of

information systems security can be dealt with through well established reliability methods. This approach, if successful, would also assist in the application of quantitative methods in the evaluation of information systems security. Alternative methods are likely to be required, especially action research as suggested by Baskerville (1994), but disciplining the research in information systems security is fundamental for both scientific study and development of secure systems. The field of information systems research (*e.g.* Galliers 1992) should be considered as a natural framework for research in information systems security.

Establishing flexible safeguards and relationship between information systems research and research in information systems security, mechanisms may be established that integrate design of security and systems in general. While the design duality remains, security design and system design remain in conflict at least by two ways (Baskerville 1995): prevention of normal system operations by security enforcement, and prevention of innovative adaptations to unexpected situations in organisations. As flexibility appears a key success factor for defusing software crisis, high hope can be established on flexibility and integration of safeguard design enabling effective security measures.

Conclusions

A fundamental problem of systems remaining vulnerable to security violations known for decades has been studied. The cause of the problem lies deeper than in the security awareness of system designers and programmers, in the weak scientific foundation of information systems security and primitiveness of system security design methods. A number of causes of security violations suggest inappropriate understanding of concepts of information security among researchers and practitioners. Alternatively, a number of fundamentally different definitions originating from cryptographic researchers, computer security researchers, network researchers and information systems security researchers make it virtually impossible to establish theories for dealing with all facets of comprehensive security of information systems. To overcome these problems, generic research frameworks are required to establish the scientific discipline of information systems security. Otherwise, research remains fragmented and inconsistent, and security measures continue preventing normal and innovative system operations.

References

Anderson, R. (1993), Why Cryptosystems Fail. *1st ACM Conference on Computer and Communications Security*. ACM Press.

Backhouse, J. and Dhillon, G. (1996) Structures of Responsibility and Security of Information Systems. *European Journal of Information Systems* 5(1):2-9.

Baskerville, R. (1991) Risk Analysis: An Interpretative Feasibility Tool in Justifying Information Systems Security. *European Journal of Information Systems* 1(2):121-130.

Baskerville, R. (1992) The Development Duality of Information Systems Security. *Journal of Management Systems* 4(1):1-12.

Baskerville, R (1993a) Information Systems Security Design Methods: Implications for Information Systems Development. *ACM Computing Surveys* 25(4):375-414.

Baskerville, R. (1993b) Information Systems Security: Adapting to Survive. *Information Systems Security* 2(1):40-47.

Baskerville, R. (1994) Research Notes: Research Directions in Information Systems Security. *International Journal of Information Management* 14(5):385-387.

Baskerville, R. (1995) The Second-Order Security Dilemma. In W. Orlikowski, G. Walsham, M. Jones & DeGross, J. (eds) *Information Technology and Changes in Organisational Work* (pp.239-249). Chapmann & Hall, London, UK.

Bell, D. E. (1998) Concerning "Modelling" of Computer Security. *IEEE Symposium on Security and Privacy*, pp. 8-13. IEEE Computer Society Press.

Blakley, B. (1996) The Emperor's Old Armor. *ACM New Security Paradigms Workshop*, pp.2-16. ACM Press.

Bleinbacher, D. (1998) Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS#1. *Advances in Cryptology - CRYPTO'98*, pp.1-12. Springer-Verlag LNCS 1462.

Boogaard, M. (1994) *Defusing the Software Crisis: Information Systems Flexibility Through Data Independence*. Thesis Publishers, Tinbergen Institute Research Series 79. Amsterdam, The Netherlands.

Borenstein, N.S., Ferguson, J., Hall, J., Lowery, C., Mintz, R., Morris J., New, D., Parenti, B., Rose, M., Steffurd E., Stein, L., Storm, C., Vielmetti, E., Weiser, M. and Wolff, P.-R. (1996) Perils and Pitfalls of Practical Cybercommerce. *Communications of the ACM* 39(6):36-44, June.

Boswell, A. (1995) Specification and Validation of a Security Policy Model. *IEEE Transactions on Software Engineering* 21(2):63-68.

Burrows, M., Abadi, M. & Needham, R. (1990) A Logic of Authentication. *ACM Transactions on Computer Systems* 8(1):18-36.

Cheswick, W.R. & Bellovin, S.M. (1994) *Firewalls and Internet Security, Repelling the Wily Hacker*. Addison Wesley, Reading, MA, USA.

Common Criteria for Information Technology Security Evaluation v2.0, parts 1-3 (1998) International Standard ISO/IEC 15408, CCIB-98-026.

Data Encryption Standard, DES (1977) Federal Information Processing Standards Publications (FIPS PUB) 46. National Bureau of Standards, Jan.

Denning, D.E.(1999) *Information Warfare and Security*. ACM Press, Addison-Wesley, Reading, MA, USA.

Diffie, W. & Hellman, M. E. (1976) New Directions in Cryptography. *IEEE Transactions on Information Theory* 22(6):644-654.

Escamilla,T. (1998) *Intrusion Detection. Network Security Beyong the Firewall*. John Wiley & Sons, Inc. NY, USA.

Fitzgerald, G. (1990) Achieving Flexible Information Systems: The Case of Improved Analysis. *Journal of Information Technology* 1990(5):5-11.

Fried, L. (1994) Information Security and New Technology: Potential Threats and Solutions. *Information Systems Management* 11(3):57-63.

Galliers, R., ed. (1992) *Information Systems Research : Issues, Methods, and Practical Guidelines*. Blackwell Scientifc Publications, Oxford, England.

Hartmann, A. (1995) Comprehensive Information Technology Security: A New Approach to Respond Ethical and Social Issues Surrounding Information Security in the 21st Century. *IFIP TC11 11th International Conference of Information Security*, pp.590-601. Chapmann & Hall.

Jonsson, E. (1998) A Novel Approach to Security and Dependability Concepts and Measures. *3rd Nordic Workshop on Secure IT Systems.*

Kaijser, P. (1998) A Review of the SESAME Development. *3rd Australasian Conference on Information Security and Privacy*, pp.1-8. Springer-Verlag LNCS 1438.

Kowalski, S. (1990) Computer Ethics and Computer Abuse: A Longitudinal Study of

Swedish University Students. *IFIP TC11 6th International Conference on Information Security*.

Kwok, L. and Longley, D. (1997) Code of Practise: A Standard for Information Security Management. *IFIP TC11 13th International Conference on Information Systems Security*, pp.78-90. Chapman & Hall.

Landwehr, C. E., Bull, A. R., McDermott, J. P. & Choi, W. S. (1994) A Taxonomy of Computer Program Security Flaws. *ACM Computing Surveys* 26(3):211-254.

Leiwo, J. & Heikkuri, S. (1998) A Group-Enhanced ISSI Model for Secure Interconnection of Information Systems. *IFIP TC11 14th International Conference on Information Systems Security*, pp.271-282.

Leiwo, J., Gamage, C. & Zheng, Y. (1999) Harmonisation of Information Security Requirements. *Informatica* 17 (to appear).

Linn, J. (1997) *Generic Security Service Application Program Interface, Version 2*. IETF RFC2078.

Matsui, M. & Yamagishi, A. (1992) A New Method for Known Plaintext Attack of FEAL Cipher. *Advances in Cryptology - Eurocrypt'92*, pp.81-91. Springer-Verlag.

Murphy, S. (1990) The Cryptanalysis of FEAL-4 With 20 Chosen Plaintexts. *Journal of Cryptology* 2(3):145-154.

Neumann, P. G. (1995) *Computer Related Risks*. ACM Press, Addison-Wesley, NY, USA.

Pfleeger, C. P. (1997) *Security in Computing*, 2nd ed. Prentice Hall, Upper Saddle River, NJ, USA.

Pressman, R. S. (1997) *Software Engineering: A Practitioner's Approach*, 4th ed. McGraw-Hill, NY, USA.

Schneier, B. (1996) *Applied Cryptography*, 2nd ed. John Wiley & Sons, NY, USA.

Schneier, B. & Mudge [sic] (1998) Cryptanalysis of Microsoft’s Point-to-Point Tunneling Protocol (PPTP). *5th ACM Conference on Communications and Computer Security*, ACM Press.

Shannon, C. E. (1949) Communication Theory of Secrecy Systems. *Bell System Technical Journal* 28:50-64.

Shoup, V. (1998) *Why Chosen Ciphertext Security Matters*. IBM Research Division, Computer Science/Mathematics Technical Report RZ 3076. Zurich, Switzerland, Nov.

Spafford, E. H. (1989) The Internet Worm Program: An Analysis. *Computer Communications Review* 19(1):17-57.

vonSolms, R. (1996) Information Security Management: The Second Generation. *Computers & Security* 15(4):281-288.

vonSolms, R. & Meyer, L. (1995) Information Security Accreditation - The ISO9000 route. *IFIP TC11 11th International Conference on Information Systems Security*. Chapman & Hall.

Steiner, J. G. and Neumann, C. and Schiller, J. I. (1988) Kerberos: An Authentication Service for Open Network Systems. *Winter USENIX Conference*, pp.191-202. USENIX Association.

US Dept. Defence Standard Defence System Software Development DOD-STD-2167A (1998). Feb. 29.

*US National Security Agency SSE CMM: Systems Security Engineering Capability Maturity Mode*l (1995) SSE-CMM Model and Application Report. Oct. 2.

Williams, J.G. & Abrams, M.D.(1995) Formal Methods and Models, pp.170-186. In Abrams, M.D., Jajodia, S. & Podell, H.J., eds., *Information Security, An Integrated Collection of Essays*. IEEE Computer Society Press, Los Alamitos, CA, USA.