INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI®

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

A Commercially Viable Computer Security Implementation Framework

Marc Bouffard

A Thesis

in

The Faculty

of

Commerce and Administration

Presented in Partial Fulfilment of the Requirements
for the Degree of Master of Science in Administration at
Concordia University
Montreal, Quebec, Canada

March 1998

Abstract

A Commercially Viable Computer Security Implementation Framework

Marc Bouffard

Commercial computer security concerns have grown in importance with the continued rise of computer literacy among the general populace. Despite this, the education of management information system professionals in the application of computer security techniques has been largely ignored. This study groups a comprehensive list of security methods using Leonard Fine's *Total Computer Security Concept and Security Policy*, which divides security concerns into 9 categories: security policy, organization, physical and fire, personnel practices, insurance, systems security, application security, standards and the audit role. Due to the vast scope of the framework, only one of Fine's categories was validated: systems security. To allow computer security issues to be addressed in a timely manner, the implementation schedule of each method has been addressed in terms of a 4 phase Systems Development Life Cycle. Finally, to address commercial security concerns, a third dimension of cost/effectiveness was added for each method under consideration. The conclusions include the results of the validation of the 12 systems security methods, as well as further research possibilities.

Dedication

I would like to thank Dr. Jamshid Etezadi and Dr. Jerry Tomberlin for their support throughout this prolonged research. In addition I would like to thank my family for encouraging me into the graduate program and supporting me during my stay here, and Chris "Narc" Beck and Bruno Gonzalez for their support, encouragement and, above all, friendship. Finally, I would like to thank my nephew, Jordan Marsh, for thinking that this thesis was "cool".

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Commercial computer security has been an important and yet often overlooked area in the field of management information systems. Few information systems texts mention computer security more than in passing and yet IS professionals are responsible for designing and implementing many of the systems in which security is a critical element.

Models in common use such as the US Department of Defense's *Trusted Computer System Evaluation Criteria* (DOD, 1983) and the *Canadian Trusted Computer Product Evaluation Criteria* (CSSC, 1993), while extensive, are practical only from a military perspective. Widely respected for their contribution to information safeguarding, their implementation is difficult and expensive when applied to a commercial environment.

Specifically, these models seek to regulate the control of classified information. While this is an important goal from a military standpoint, data integrity is more important to commercial organizations (Clark and Wilson, 1987). In addition, the cost of these military measures is difficult to justify in a commercial environment, and often simply too expensive for a commercial organization to implement (Chalmers, 1986).

Other models designed to be applied in more general cases such as Donn Parker's *Computer Security Techniques* (Parker, 1982) and Ruder and Madden's *Analysis of*

Computer Security Safeguards for Detecting and Preventing Intentional Computer Misuse (Ruder and Madden, 1978) are structured as a series of methods targeted at certain functions such as user authentication, and overall system security. While critical, these functions are difficult to apply as they do not correspond to well known information systems methodologies.

When looking away from general models, two main areas in computer security have been addressed: technical measures and organizational issues. The former has been dominated by computer science techniques such as encryption algorithms and password schemes (Farmer, Venema and Wietse, 1993; Blanton, Ellis and Rosenburg, 1991). The latter is more in the domain of the IS professional and therefore is more apt for application as a basis for, or element in, a general computer security plan.

Much of the research done in the organizational field has either centred around the discovery and prevention of computer abuse (Straub and Nance, 1990; Straub, 1990; Lee, Segal and Steier, 1986; Mylott, 1985; Parker, 1975) or the management of the security function in an organization (Hoffer and Straub, 1989; Wong, 1987; Wong, 1986). While these areas are of critical importance to an overall security plan, they are still but a small part of the greater picture. Some others address the implementation of computer security techniques in very specific environments such as micro-mainframe networks (Bookholdt, 1989) or local area networks (Jamieson and Low, 1989) and while these can be used as implementation guides, their use is obviously limited to their areas of concern.

In addition, security methods should be implemented as the system is implemented rather than as an afterthought. "An existing system is not normally modified to make it secure. The accepted way of acquiring a reasonable assurance that the system is secure is to design and build it according to some agreed upon model. If the model can be proved secure and the system can be proved to follow the model, then we have increased the ability of the system to safeguard the information entrusted to it"(Vaughn, Saiedan & Unger, 1993).

The goal of this framework is to provide a general guideline for any commercial organization wishing to evaluate its computer security needs, and implement security methods, in a timely manner, to offset risks. In order to accomplish such a goal, the framework has three dimensions of interest: 1) *Fine's Total Computer Security Concept and Security Policy* which subdivides global security concerns into 9 distinct categories, 2) the systems development life cycle which allows us to address implementation timing issues and, finally, 3) cost.

Fine's Total Computer Security Concept and Security Policy

The first of these dimensions is Leonard Fine's *Total Computer Security Concept and Security Policy* (Fine, 1982; Fine, 1978). For any security framework to be effective, it must categorize security into smaller sections in order to make it feasible to implement (von Solms, van de Haar, von Solms and Caelli, 1994). The classification that Fine put forward in this model breaks down into 9 distinct categories or "pillars", these are:

1. Security policy
2. Organization
3. Physical and Fire
4. Personnel Practices
5. Insurance
6. System Security
7. Application Security
8. Standards
9. Audit Role

Fine visualized computer security as a physical structure. With the above "pillars" supporting a beam called the "Total Computer Security Concept".

1. The security policy includes an overall, management supported, security procedure and policy. The appointment of a computer security manager to oversee the implementation and maintenance of security areas (in this case, the pillars) is critical. (Wong, 1986)

Elements that the policy should include are: assets accountability assignment (Parker,0 1982), a disaster recovery plan and test schedule (Hoffer and Straub, 1989; Parker, 1982), a user agreement (Parker, 1982), an end user education plan (Hoffer and Straub, 1989; Courtney, 1986), an employee termination policy (Ruder and Madden, 1978), a policy for sensitive information retrieval and dispersal (CSSC, 1993; Wong, 1987; DOD, 1983; Fine, 1982; Parker, 1982; Fine, 1978; Ruder and Madden, 1978), object reuse policy (CSSC, 1993; NCSC, 1992; DOD, 1983;), classification of access rights by job function (Parker, 1982) and computer use access control administration (Parker, 1982).

2. Organization includes the separation of duties throughout the organization into logical functions. This is the first step of the Clark-Wilson data integrity model (Clark and Wilson, 1987) which was developed as the commercial alternative to the military models. To increase the effectiveness of separation of duty as a security tool, job rotation should also be introduced (Ruder and Madden, 1978). This reduces the possibility of collusion which, if it occurs, could seriously undermine the effectiveness of separation of duties.

Of particular interest is the separation of sensitive jobs from the bulk of other jobs. Sensitive jobs should be executed on trusted machines by trusted individuals. The attention paid to the security of these jobs will reduce risk to a manageable level for the most critical jobs in the organization (Parker, 1982).

Finally, the establishment of a security committee as a unique governing body of computer security is necessary as the responsibility for implementing the security policy is theirs (Wong, 1986; Ruder and Madden, 1978). It is important that a high level executive is a member of, or consulting closely with, the security committee who must have the authority and upper management backing for them to be effective in implementing tools and policies and procedures, and to realistically discipline those that do not follow the guidelines set forth (Wong, 1986).

3. Physical and Fire is defined as the physical security surrounding the actual computer system as well as all related work areas. Precautions include: a physical security perimeter and access barriers (Parker, 1982; Ruder and Madden, 1978), physical personnel authentication (Parker, 1982), a regular backup schedule (Ruder and Madden, 1978), electrical power shutdown, recovery and safeguards (Parker, 1982), fire detection and extinguishment (Ruder and Madden, 1978), computer inventory controls (Ruder and Madden, 1978), alternate communication paths for critical online systems (Ruder and Madden, 1978), and computer terminal access and use restrictions (Parker, 1982). It is the straightforward, nuts and bolts of physical prevention techniques.

Physical security once defined computer security (Loch, Carr and Warkentin, 1992; Fine, 1978). Its importance, although still significant, is waning as other issues such as

telecommunications and networks come further into the forefront (Loch, Carr and Warkentin, 1992).

4. Personnel Practices provide guidelines for the establishment of computer personnel and personnel practices for the organization. End user education and the end user agreement share a dual life, in that they could also be included in this category. They were included as part of the security policy to reflect their importance in the regulating and educating of users in proper and secure behaviour on the system.

This category concentrates on personnel assignment in an organization to manage and assure a secure computing environment. This includes the establishment of a computer security officer, a computer user coordinator and an EDP (Electronic Data Processing) auditor (Parker, 1982). The computer security officer is responsible for the establishment and management of computer security throughout the organization. The computer user coordinator manages all computer user issues, including questions, change of access rights and receiving trouble calls. Finally, the EDP auditor periodically verifies all elements of computer security to insure their proper functioning.

The computer user coordinator is also responsible for computer user trouble calls (Parker, 1982) which help define the policies and procedures that should be followed by

the computer security officer, the computer user coordinator and all system users whenever a problem is discovered and reported.

Finally, the cooperation of computer security officers of various organizations can help increase the computer security levels of all businesses involved (Parker, 1982).

5. Insurance covers any financial precautions against security risks taken by a company. This is fairly straightforward and includes not only actual insurance policies purchased in case of disaster. It also includes external service contracts in case of a general system failure, thus allowing computer operations to continue even if the computer system meets a fatal end. This stage is all the more important today, when the loss of data or system time can be crippling to a company that depends heavily on it (Loch, Carr and Warkentin, 1992).

6. System Security includes all system-wide, application independent software and hardware based security methods, including network security. These issues include: a network firewall to protect organizational data from external threats, remote terminal physical security (Parker, 1982), restriction of systems utility programs (Parker, 1982; Ruder and Madden, 1978), file and/or program assignments (Parker, 1982), data classification (Parker, 1982; Ruder and Madden, 1978), the Bell-Lapadula data rule (McHugh and Thuraisingham, 1988), technical reviews of operating system changes (Parker, 1982), cryptographic protection (Parker, 1982; Ruder and Madden, 1978),

user authentication (Parker, 1982; Ruder and Madden, 1978), automatic, timed, terminal logoff (Ruder and Madden, 1978), hardware monitors (Ruder and Madden, 1978) and bill back systems (Ruder and Madden, 1978).

Managing an operating system means keeping up with the latest threats to each computer system (Wong, 1986), so important is this function that an external body was implemented to help diffuse system security threats. The Computer Emergency Response Team (CERT) is a body of individuals concerned with informing the interconnected public of security risks that could affect them. While it was meant principally for those who had systems connected to the Internet, their advisories explain operating system or other common security weaknesses that can be abused. These CERT advisories are available to any who wish them simply by sending email to cert@cert.sei.cmu.edu and asking to receive them. They are issued whenever they are necessary and not only warn users of possible security risks, but also indicate exactly how to correct the problems.

7. Application Security to assure that source code and data files only be modified by authorized users, and that these changes conform with security standards. This is a critical function and along with it goes the added burden of only allowing authorized users access to applications and data files (Thuraisingham and Rubinovitz, 1992; Thuraisingham 1991; McHugh and Thuraisingham, 1988)

This can be a very difficult task, and since the data on a computer system is often its most valuable resource (Wong, 1987), a critical one. In today's distributed environment, it is not uncommon that a user download some data for local processing or analysis. Once this data leaves the confines of the corporate database, its security can often quite easily be compromised (Wong, 1987; Mylott, 1985). The security aspect of distributed databases is currently a very hot issue (Thuraisingham and Rubinovitz, 1992; Thuraisingham,1991; Goyal and Singh, 1991; Laferriere, 1990; McHugh and Thuraisingham, 1988) . Corporate data is the lifeblood of the organization, and making it tamper proof should be of the highest concern. The saying that "Information is power" was born upon this realization.

To this end application and data security methods include: production program authorized version validation (Parker, 1982), responsibility assignment for application program controls (Parker, 1982), program quality assurance (Parker, 1982; Ruder and Madden, 1978), program change logs (Parker, 1982), secrecy of data file and application names (Parker, 1982), programming library access controls (Parker, 1982), input data validation (Parker, 1982), data file access controls by subfunction (Parker, 1982), application program testing policy (Ruder and Madden, 1978), initial program load (IPL) checks (Ruder and Madden, 1978), processing time controls (Ruder and Madden, 1978) and creating well formed transactions (Clark and Wilson, 1987).

8. Standards for applications development, systems development and controls need to be defined and implemented early in the computer systems development. The standards define certain basic efforts that must be taken to assure a minimum level of quality, security and legal compliance.

They include: compliance with laws and regulations (Parker, 1982), participation of users at critical development times (Parker, 1982), application system design verification (Ruder and Madden, 1978) and application program standards (Ruder and Madden, 1978).

9. The Audit Role is a complex issue. It involves the methods by which the system is periodically evaluated for unusual occurrences, security breaches and continued security worthiness (Straub and Nance, 1990). All manner of audit logs are possible. Sensitive file modification logs, crash audit logs, application program change control log, operator console log and improper logon log. In addition to logs, other internal procedures include independent computer use by auditors (Parker, 1982), requirement and specification participation by EDP auditors (Parker, 1982), exception reporting (Parker, 1982), monitoring computer use (Parker, 1982), periodic computer resource audits (Ruder and Madden, 1978) and independent control of audit tools (Parker, 1982).

In a recent article, Straub and Nance (Straub and Nance, 1990) ascertained that 41% of computer abuse incidents were discovered by accident, 50% were discovered by systems controls and only 16% were discovered by active detection. From these figures it become evident that audit logs and other systems controls play a very important role in the discovery of computer abuse. Their maintenance and use should be of primary concern.

External auditing specialists should also be called in periodically (Straub and Nance, 1990). This will help avoid computer security risks stemming from the organization's security committee or audit team. Unfortunately, if sensitive data exists on the system, it may be undesirable to have external audits.

Fine's 9 "pillars" provide a simple, complete and effective way of classifying security methods by their purpose. It also provides a guideline which security personnel can use when determining what methods to implement in their organization.

Systems Development Life Cycle

The systems development life cycle is one of many methods used to develop a computer system. It has been argued by some that the SDLC has become obsolete, but it is still one of the main methodologies taught in university information systems curricula. The solid structure and ease of use of the SDLC lends itself particularly to a security framework, allowing an easy to use and effective method to determine proper timing of security method implementation.

This does not mean that a system must be developed using the SDLC for this framework to prove effective. On the contrary, many methods such as phased commitment, evolutionary, prototyping and end-user development (Laudon and Laudon, 1988) could be used in place of SDLC, but their structures do not lend themselves to the issues of structured timing and sequencing.

The SDLC will allow a system designer to know when a methodology should be applied to the system, and parallels can easily be inferred between the different stages of the SDLC and other formal methods of systems analysis and design.

Adding further complication to the use of the SDLC is the lack of standardization in this area. The SDLC can vary from as many as nine steps (Gibson and Hughes, 1994): problem definition, systems analysis, systems design, system development, system

testing, system implementation, formal review, system project modification and enhancement and system maintenance to as few as four main steps (Whitten, Bentley and Barlow, 1989): systems analysis, systems design, systems implementation and systems support.

For ease of use, I have adopted the 4 step systems development life cycle developed by Whitten, Bentley and Barlow, (1989): system analysis, system design, system implementation and system maintenance (or support). This reduces the complexity of the SDLC, limiting it to the minimum number of steps feasible in systems development, while allowing the timing issues needed for the framework to be addressed.

1. System Analysis is the study of a current business system and its problems, the definition of business needs and requirements, and the evaluation of alternative solutions. (Whitten, Bentley and Barlow, 1989)

2. System Design is the general and detailed specification of a computer-based solution that was selected during systems analysis (Whitten, Bentley and Barlow, 1989). This includes hardware and software purchasing or development decisions.

3. System Implementation is placing the system into operation. Computer programs are written and tested, managers and users are trained to use the new system, and operations are converted to the new system. (Whitten, Bentley and Barlow, 1989)

4. System Maintenance (or support) is the ongoing support of the system after it has been placed into operation. This includes program maintenance and system improvement (Whitten, Bentley and Barlow, 1989).

These stages exist in most methodologies, though not necessarily in the order defined by the SDLC. They describe the necessary steps that will be completed when developing any information system.

Cost

The cost dimension is perhaps the most important in the commercial environment and yet, it is often overlooked in current computer security models. The reasons for this is that most computer security models in use today were designed for use in military environments. As stated, the goal of a military computer security model is to protect their information from the enemy at all costs. Private sector organizations are more interested in data integrity than data privacy (Chalmers, 1986). Corporate system administrators, though willing to implement computer security on their systems, find that cost justification using such models is difficult.

Traditionally, if a system is purchased, the hardware costs are often used as the basis for security justification (Wong, 1987). However, with the high power and low cost of today's computer systems, it becomes difficult to justify the cost of computer security solely on the cost of hardware (Wong, 1987).

Unfortunately, the hardware costs are usually a fraction of the value of the data and programs that are contained on the system (Wong, 1987). It has been suggested that the justification for system security not come from the value of the hardware, but rather the value of the applications and data which are guarded by the security techniques (Wong, 1987). With this development another problem emerges: how to value the data? Past subjective valuation of data has proven to be highly unreliable. In a famous

hacker case, Southern Bell evaluated a 12 page document simply called "E911" which outlined their emergency 911 services as being worth $79,449.

The breakdown of this figure follows (Sterling, 1992):

- technical writing: 200 hours @ $35/hour:	$7,000
- project manager to oversee the technical writer 200 hours @ $31/hour:	$6,200
- a week of typing:	$721
- a week of formatting:	$721
- a week of graphics formatting:	$742
- 2 days of editing:	$367
- box of order labels:	$5
- preparing the purchase document:	$129
- printing	$313
- placing document in an index: 2 clerks for one hour each:	$43
- VT220 terminal(used to type document)	$850
- VAXstation II(computer system)	$31,000
- Printer	$6,000
- Interleaf software	$22,000
- VMS software	$2,000

All this to write a 12 page document which was sold in retail outlets for the price of about $20 U.S. Preposterously enough, the hardware costs were included in the valuation of the document. As you can see, management cannot be trusted to estimate the value of their own data. An objective scheme to estimate the value of data needs to be developed before the value of the data could be used as the basis for determining the computer security budget (Fine, 1982).

Although the development of such a valuation scheme is beyond the scope of this framework, the relative cost of each method has been listed to allow system

administrators and other corporate computer security personnel the chance to evaluate each method based on their cost as well as their effectiveness as security tools. Using these costs, private sector companies will be able to implement security using, in most cases, cost as a justifier. It is true that some tools are indispensable for a proper computer security plan (eg. backups), but other tools may overlap each other, and cost can prove to be a valuable criteria in determining which one will be implemented.

Risk Analysis

Before the implementation of the following framework can be attempted, a risk analysis of the organization should be conducted. Formal risk analysis can involve many steps, and the methodologies range from mathematical models to subjective analysis by computer security personnel (Perry and Kuong, 1981).

Unfortunately, due to the high level of uncertainty involved in evaluating risk levels, mathematical models have proven difficult to use, and often are no more accurate than subjective models.

Risk itself is a complex issue. It is defined as the likelihood of a potential threat causing an adverse consequence to the organization (Perry and Kuong, 1981). Apart from physical threats, such as fire or disaster, there is very little historical information that will help determine the risk levels in an organization. Despite this, risk must be evaluated in order to determine what areas of the organization are security concerns.

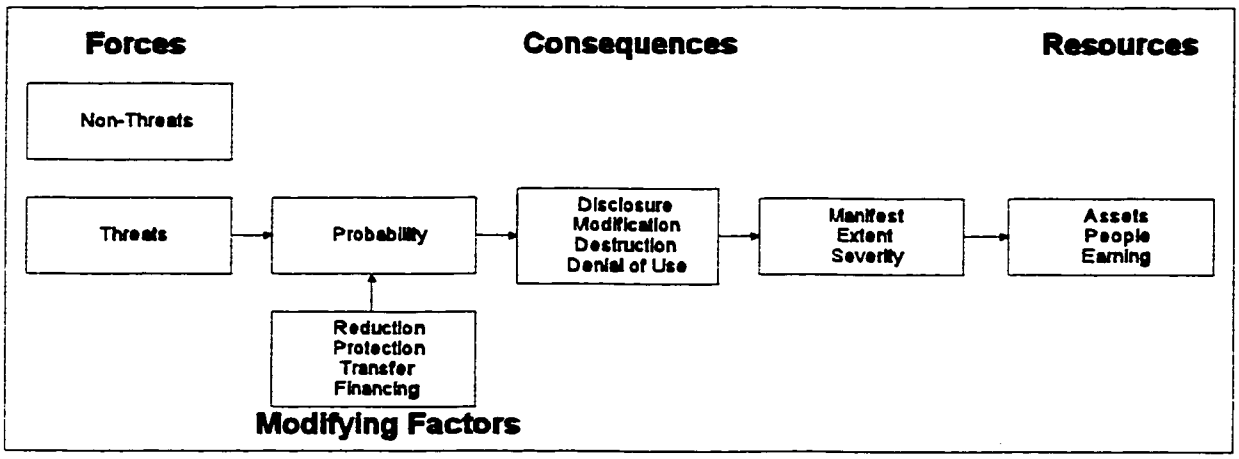


Figure 1. The Components of Risk (Loch, Carr and Warkentin, 1992)

Risk is composed of three main factors: forces, consequences and the resources affected. Forces include any threats to the organization. These include, but are not limited to: viruses, hackers, fire and other disasters, employee accident and malicious actions, power failures etc. The list of threats to any organization is extensive, and non-generalizable. Each organization has threats particular to it, based on competition, geographic location, past history, size and other factors.

The consequences of these threats range from disclosure of information to denial of system use. Risk analysis attempts to identify these threats and, once identified, security tools can be applied to reduce the probability (or risk) that these threats will result in negative consequences to the organization.

A simple risk analysis model is composed of 4 steps:

1. List all potential risks.
2. Determine the probability of each risk affecting the company.
3. Estimate the cost of the consequences associated with each of these risks.
4. Rank each threat according to their importance to the organization.

Steps 2 and 3 of this model will, in all likelihood, have to be estimated to the best of the security personnel's ability (Courtney, 1986). Any available historical information should be used in the valuation of probabilities and costs.

Each of these four steps should be conducted for each of Fine's 9 classifications. The added structure will not only simplify the risk analysis task, but will also facilitate the implementation of this model to specific organizational security needs.

Levels of acceptable risk should be determined for each classification. It is impossible to offset all risk, and it would be far too expensive to try. Security methods should be implemented to achieve this level of acceptable risk, and not to offset all risk.

The results of the risk analysis can be used not only to determine what security methods to implement, but also to justify the controls chosen, and indeed the entire security project, to upper management. Risk analysis will also influence the audit plan, which should target the areas most vulnerable to risk (Perry and Kuong, 1981).

Once the areas most vulnerable to risk have been ascertained, the framework is used to determine what security measures are available to reduce the risk.

Chapter 2

A Commercially Viable Computer Security Implementation Framework

This framework was designed specifically with the information system professional in mind. Applied following a risk assessment of the organization, it allows security methods to be implemented as the system is being developed. In this manner, risk levels are addressed at the outset of system design. The security of the resulting system will thus offset threats to the degree necessary to achieve acceptable levels of risk., but no more. By avoiding excessive levels of security, unnecessary losses in user productivity can likewise be avoided.

Each method has been assigned to one of the nine categories detailed in Fine's *Total Computer Security Concept and Security Policy* (Fine, 1982). A cost has been included for each method under consideration, where available. Cost has been described in 2 ways. Where the cost of the method has been ascertained from the literature review, it has been described using the relative scale, either Low, Medium or High. For those methods grouped within the Systems Security category, cost has been listed as a ratio of Cost/Effectiveness determined from the results of the survey. A value of less than one indicating that the method under consideration is more effective than it is costly.

Finally, each method is described and includes an implementation schedule following the four stage Systems Development Life Cycle (SDLC). Only those stages of the SDLC which are of importance to the method are detailed.

Security Policy

Method	Description	Cost
Assets Accountability Assignment (Parker, 1982)	Assigning responsibility for upkeep, maintenance and safety of all hardware elements.	
Disaster Recovery (Hoffer and Straub, 1989; Parker, 1982)	The policy must include a comprehensive disaster recovery plan.	
User Agreement (Parker, 1982)	Each user who will access the system must sign an agreement to use the system in specific ways.	
End User Education (Hoffer and Straub, 1989; Courtney, 1986)	A comprehensive end-user education plan to give users the knowledge they need to operate in a secure manner.	
Employee Termination Policy (Ruder and Madden, 1978)	A policy designed to limit the damage done by a disgruntled and terminated employee.	
Object Reuse(CSSC, 1993; NCSC, 1992; DOD, 1983)	Guidelines to be followed when reusing any object (magnetic or optical media, memory area, page frame...) that once contained sensitive information. The information must be removed entirely before being reused.	
Classification of Access Rights by Job Function	Each user's access and privileges on the system should be determined by their job description. Homogeneity of permissions within job functions will lead to ease of administration and accountability.	
Disciplinary Actions (Wong, 1987)	The policies by which security offenders will be held accountable for their actions.	
Computer Use Access Control Administration (Parker, 1982)	Establishing the process by which access rights can be modified.	

Classification: Security Policy

Method: Asset Accountability Assignment

Description: The methodology for the assignment of responsibility for assets is determined. Company assets are assigned to those who use them, and users are informed of their responsibilities towards the assets assigned to them.

Managers are informed of what assets they, and their staff, are responsible for. These responsibilities should be clearly and explicitly defined, allowing no areas of ignorance or confusion.

All responsibilities assigned should be discrete and self-contained. No responsibility overlaps should be permitted. The responsibility for the correct and secure workings of any asset should be assigned to only one person. That person should be held responsible for any damage caused to the equipment, any misuse of equipment or any failure in their responsibility for correct and secure operations.

Asset responsibility includes proper care and maintenance of the asset. Should the asset fail to meet it's established requirements, the person who is responsible for that asset should take proper measures to see that the asset is replaced or repaired.

Asset accountability lowers the levels of trust needed for each manager, as the responsibilities for proper operations in their department is shared by all employees of that department.

The asset accountability for each area of the enterprise should be well documented and kept up to date. Any misuse of an asset should lead to immediate and appropriate disciplinary action for the employee responsible for that asset.

Analysis
The general asset accountability rules are determined. Employees are segregated according to their job functions (clerks, managers, operations etc.) and general guidelines are developed in order to delineate what asset types will be assigned to which job functions. For example, personal computers responsibility will be assigned to the clerk using it, while responsibility for the main server will be assigned to the operations group.

Design
Once the specific choice of hardware is selected and the equipment is purchased, appropriate identification should be affixed to each piece of equipment (see Physical and Fire: Computer Inventory Control). The accountability for each piece of equipment should then be assigned to the appropriate employee. The initial assets accountability document should be completed in this stage.

Implementation
As assets are implemented, and before the system becomes operational, all affected employees should be informed as to their responsibilities towards the equipment. Each employee's responsibility should be made clear to them and disciplinary actions taken due to a failure in meeting their responsibilities should be clearly explained. The reasons for the division of assets responsibilities should be stated, as well as to whom the employees can report any misuse or malfunction of assets.

Maintenance
The asset accountability list should be kept up to date, with periodical auditing of the list ensuring its completeness and correctness. The list should be updated whenever new employees are hired, new assets are purchased, an employee is terminated or changes function, or an asset is replaced.

Classification: Security Policy

Method: Disaster Recovery

Description: A disaster recovery plan is a plan for emergency response, backup operations and post-disaster recovery maintained by an activity as a part of its security program that will ensure the availability of critical resources and facilitate the continuity of operations in an emergency situation.(NCSC, 1988)

A disaster recovery plan must be formally written, updated, tested and audited to ensure it's completeness and correctness in the case of an emergency. Personnel involved **must** be familiar with their role in such a plan, and test runs are strongly suggested to ensure preparedness in case of an emergency. The presence of a documented disaster recovery plan should not lead to complacency in this area.

Although it is possible to develop a generic disaster recovery plan, it is often necessary for each plan to be customized according to geographical location, operational parameters, specific computer systems, and local or environmental hazards. It must be ensured to include all potential disasters.

The aspects of such a plan include, but are not limited to: coordination, systems and support, hardware recovery or replacement (see Physical and Fire: Contingency Recovery Equipment Replacement), facilities, administration, scheduling, communications, documentation, supplies, insurance. Each aspect of a disaster

recovery plan should be assigned to an individual. It will then fall to this individual to ensure that the aspect of the plan for which they are responsible be accomplished and properly documented for future reference.

Systems, applications and functions should be prioritized in order to determine their precedence in disaster recovery. A liaison with users should be established in order to facilitate the execution of the plan. Communication is a critical part of any disaster recovery plan, particularly in large organizations. Communication between managers can help to assure a smooth and minimal cost recovery.

Once a disaster recovery plan has been used, all affected job functions should be monitored for any change in productivity. The time period of productivity fluctuation as well as the reason for these fluctuations should be documented. These documents, as well as any historical disaster recovery documents, should be analyzed to determine how the plan can be improved.

Analysis
If no documented disaster recovery plan currently exists for the organization in question, then one should be developed before any planning for new systems continue. If the plan does exists, then it must be modified to include the new system. All elements of the plan must be reviewed and, if necessary, modified to include the new planned system. The system should be designed with all possible disasters in mind. Individuals responsible for aspects of the disaster recovery plan should be consulted in order to determine the best design for the new system in these respects. Any historical disaster recovery documentation must be consulted in order to determine if any improvements to the system will aid in future disaster recovery attempts.

Design
The system is designed with all precautions necessary to survive a potential disaster. The disaster recovery document is officially modified in order to meet any new requirements that result from the implementation of the new computer system. Elements of the disaster recovery plan such as alternate communication paths (see Physical and Fire: Alternate Communications Path), as well as any improvements identified by the review of historical documents are included in the design of the system. Any personnel responsible for specific aspects of the plan are notified of any modifications. Any emergency replacement equipment or vendor assurances as to replacement equipment are obtained.

Implementation
The plan is completed before the system goes online. All personnel involved with the plan are made aware of their roles.

Maintenance
Any modifications to the system will result in a review of the applicability of the disaster recovery plan.

Classification: Security Policy

Method: User Agreement

Description: The agreement stipulates the rules and regulations that must be followed for proper and secure use of the system.

This agreement should be designed by the computer security officer (see Personnel Practices: Computer Security Officer), the systems design staff, the computer operations group and the manager of every department. It will delineate all of the user's rights and responsibilities when using the system. The signing of this agreement should be considered legally binding, and the users should be expected to behave in a manner that is within the boundaries of proper behaviour set forth in the agreement.

The use of a user agreement serves to resolve any disputes that may arise between users and providers. Legal council may be useful in drafting such an agreement and will aid in assuring its legality. It may have to be rewritten several times before all people affected by it are satisfied. The final draft should reflect all current legislation affecting a computerized work environment, as well as all elements of the security policy relating to the users. The agreement should be drafted in such a way as to minimize it's effect on user productivity.

A user agreement will make user behaviour on the system more predictable, thus facilitating the discovery of unauthorized intruders by security staff. It also allows

providers to sensitize users to security and privacy concerns **before** the system goes online.

The user agreement should include appropriate behavior for each user based on his/her job function. This includes not only behavior on the system, but behavior in regards to the system. All infractions should be listed, including disciplinary actions that will be taken for any abuse detected (see Security Policy: Disciplinary Actions).

Analysis
The user agreement is designed in parallel with the design of the system. Any unusual user behavior that is identified at the analysis stage should be documented, as well as any unwanted user behavior. User behavior should be analyzed as well in order to determine it's appropriateness within the organization.

Design
The preliminary drafts of the user agreement are completed in this stage. As users are educated in the use of the new system (see Security Policy: End User Education), they should also be made aware of the user agreement, and any disciplinary action that will be taken if the agreement is not respected. Any user feedback should serve as a review of the agreement. Modification of the user agreement continues in an iterative way, until all parties involved are satisfied with the results. Users are then asked to sign the agreement.

Maintenance
Any change in legislation regarding the safe and proper use of computer systems may lead to a change in the user agreement. The computer security officer is responsible for monitoring all legislative modifications that may lead to a change in the user agreement. If the user agreement is modified in any way, all affected users should be informed. Once the users are familiar with the changes and agree on the new user agreement, they must sign it behave within the new boundaries. All new employees should be educated and asked to sign the agreement **before** they are allowed access to any of the organizations computer system. A periodic audit of user behavior should be conducted in order to assure the organization that user behavior falls within the boundaries set forth by the user agreement.

Classification: Security Policy

Method: End User Education

Description: Long considered to be one of the best, but most often overlooked, deterrents to computer abuse or misuse (Hoffer and Straub, 1989; Courtney, 1986), end user education should be addressed throughout a system's life cycle.

It should include training in proper systems use, systems authorizations, conditions for use, methods for changing passwords and penalties for security breaches. The users should be made well aware of why such practices and policies exist and the possible outcome should such policies be violated.

Users should be encouraged to report suspected security breaches. They should also learn to appreciate that illegal access or sabotage could effect user performance and productivity as well as damage the organization as a whole. (Wong, 1987)

Analysis
The system should be designed with user behavior in mind. Proper user behavior should be documented in order to include it in the user agreement (see: Security Policy: User Agreement)

Design
Once the user agreement is drafted, users should be educated in every element of the agreement. The reasoning as well as the penalties that will be incurred for any breach of the agreement should be made clear. Users should be asked their opinions on the agreement. Users should be made fully aware of what constitutes a breach, to whom they should report any breach, and why it is important for them to do so. The agreement should be signed by users only when they understand the full impact, as well as the importance, of computer security to the organization.

Maintenance
Periodic orientation sessions should be conducted to refresh user awareness in computer security. New users should be given security training **before** they are granted access to the system. Many security breaches are caused by user carelessness or maliciousness (Loch, Carr and Warkentin, 1992). End-user education should always be an ongoing activity. Users are the first line of defense as well as the organization's greatest vulnerability on the computer security front. It has been ascertained that 43.3% of all computer security breaches have been caused by employees either through maliciousness, or by accident (Loch, Carr and Warkentin, 1992). Proper end-user education and the use of a users agreement will help to reduce this number to a manageable level.

Classification: Security Policy

Method: Employee Termination Policy

Description: Policies and procedures to immediately withdraw all access to sensitive materials, sensitive areas and all computer systems to the terminated employee. The goal of this method is to prevent all damage to the organization from a disgruntled employee. Damage can include destruction or denial of data, programs, equipment or services and any unauthorized disclosure of data or programs.

A proper employee termination policy should include, but is not limited to (Ruder and Madden, 1978):

- Eliminating the user's account
- Change any special passwords (root, network connection etc.) that the user may have had access to.
- Remove any ability to physically access the area (confiscate keys and/or change locks)

It has been ascertained that a great many computer offenders are disgruntled employees, which puts greater importance upon the proper development and use of this tool. It is suggested that the termination policy address policies and procedures to put into effect according to the job function of the employee. If the employee was a secretary, the changing of a few passwords and the confiscating of some keys may suffice. If the employee was the systems administrator, then much more drastic measures are called

for. Changing all key passwords, confiscating keys and any property that belongs to the organization and finally, a security audit of the system to make certain that all systems are functioning properly and that no "back doors" exists to be taken advantage by a disgruntled ex-employee.

These policies and procedures should be put into effect at the earliest possible instance. If a time lag occurs between the termination of an employee and the execution of the employee termination policy, an unnecessary window of vulnerability exists for the organization.

Analysis
Any existing employee termination policy is examined to determine it's suitability for the new system. The new system is designed keeping the existing policy in mind.

Design
Once the system is developed, any user that will be using the system is classified by job function. With this accomplished, the existing employee termination policy is modified (or a new termination policy is developed) with precise instructions on dealing with a terminated employee of each job function.

Maintenance
The termination policy is modified whenever a job function is added, removed or significantly modified. If a new computer system or new security practices are introduced into the organization, the employee termination policy must be reviewed in order to assure it's completeness and correctness.

Classification: Security Policy

Method: Object Reuse

Description: The reassignment of some subject of a storage medium (eg. page frame, disk sector, magnetic tape) that contained one or more objects. To be securely reassigned, no residual data can be available to the new subject through the standard system mechanism (NCSC, 1992). Note that for the purpose of this framework, only primary and secondary storage will be addressed. If the organization is concerned with object reuse at a system's level (CPU registers, floating point co-processor registers, cache memory or physical memory), then a third party trusted system can be purchased which meets the requirements laid out in (CSSC, 1993), (DOD, 1983) or (NCSC, 1992). The development of such security tools lays beyond the capabilities of most commercial establishments.

When any fixed or removable medium is reused, any information it contains and any identification labels must be eliminated. There should be no trace of what the media contained prior to its reuse.

Organizations can purchase a degausser in order to magnetically reset removable media (magnetic tapes, floppy disks etc.). Software precautions can also be used. Overwriting the contents of the media with a series of neutral characters (eg. random zeroes and ones) can easily be accomplished completely eliminating any possibility of

someone gaining access to sensitive information from old media. This is particularly useful on fixed media, such as hard disks, which are not easily degaussed.

The security policy must explicitly state when the information on the medium is to be made inaccessible. An object may be left with data long after its primary use has been completed, if it has any value as a backup of the data in question. It is safer, however, if the medium is 'reset' either through the use of a degausser or by overwriting the data, as soon as the medium no longer serves a useful purpose. All labels identifying the medium should also be removed. Once the media is considered reset, it can then be reallocated into the resource pool.

Design
The security policy is modified to include object reuse for all media used by the system in question. If no degausser is available, the organization should acquire one that meets the requirements set forth in the policy. Software is written or purchased in order to overwrite any sensitive information contained on fixed media. This software should be flexible enough to allow overwriting for a specific file, a disk sector, track, or a complete disk.

Implementation
All rules regarding object reuse are explicitly stated and the policy is officially amended to include the new rules. It is put into effect the moment the system comes online.

Maintenance
Any changes to the system will lead to a review of the object reuse criteria set forth in the security policy. Any new media should be classified, and object reuse rules added to the security policy. The object reuse rules should be periodically audited for assurance as to their proper use.

Classification: Security Policy

Method: Classification of Access Rights by Job Function

Description: A classification scheme based on job functions. Access rights on processes and objects are set according to the job function of individuals. This not only facilitates the assignment of access rights, groups are far easier to manage then individuals, but it also lends itself to the division of responsibility tool (see Organization: Division of Responsibility).

This assures that users in the same job function can access the same processes and objects, and that to accomplish a security breach, collusion will be necessary and more difficult. Privacy and confidentiality are well-preserved with this scheme, as only those in a position to see secure data will be able to access processes or object that will give them access.

A good user authentication scheme is required to reinforce this tool (see System Security: User Authentication).

Access rights should be explicitly defined by job function, and should be documented in the security policy.

Analysis
The presence of secure data and the level of security necessary on the system should be ascertained. Potential users of the system should be classified by job function.

Design
The system is designed with both strong user authentication, as well as a method to set access privileges on objects and processes. The granularity of access necessary is determined and accommodated. Access privileges for subjects need not be set if the Bell-Lapadula rules are being maintained by the system (see System Security: Bell-Lapadula Rule). Otherwise subject access privileges must be set if users have direct access to them. Users should be interviewed by job function in order to determine what objects and processes they need to have access to. Access rights should be set using the least privilege principal (see System Security: Least Privilege Principal). All access rights by job function should be clearly documented in the security policy.

Implementation
The system should be implemented with all access privileges set. User must be made aware of what they can access, and why.

Maintenance
Access rights are periodically audited in order to assure their correctness. Any new job functions or modification of job functions will lead to a review of the security policy, and the access rights for that job function. The results and purpose of modifications should be documented and immediately implemented. Users should be made aware of any modifications.

Classification: Security Policy

Method: Disciplinary Actions

Description: The security policy should indicate to employees how they will be held accountable for their actions on, and in regarding, the computer system. This includes, but is not limited to, actions regarding access, disclosure, modification, or destruction of data.(Wong, 1987)

Any disciplinary actions taken must be at least as tough as legislation(Wong, 1987). Every person using the system must be held accountable for their actions, no organizational position should make a user immune to appropriate disciplinary action. Preferential treatment for any individual should be discouraged. Such actions have a negative impact on effective deterrence, expose the organization to subsequent legal liabilities, and bring certain moral and ethical questions to the forefront(Straub and Nance, 1990).

Disciplinary actions include reporting the offender to the authorities, if the breach goes against established legislation. It has been found that fewer than 10% of abuses were reported to the authorities (Hoffer and Straub, 1989). This has been blamed on a reluctance of the organization in making their security vulnerabilities public. Tests, however, reveal that potential violators are deterred when abuse is reported and perpetrators are prosecuted(Hoffer and Straub, 1989). Simply stressing penalties for misuse is a good deterrent (Hoffer and Straub, 1989).

Harsh penalties were deemed appropriate by only 55% of organizations surveyed for incidents involving unauthorized access, whereas 86.5% of organizations were willing to impose harsh penalties for destruction of data by manipulation or by computer virus(Loch, Carr and Warkentin, 1992). Organizations are less interested in unauthorized access than in data integrity.

In order for disciplinary actions to be an effective deterrent, they must be clearly communicated. For this reason, they should be included in the User Agreement (see Security Policy: User Agreement)(Hoffer and Straub, 1989). Actions such as disregarding the abuse, or promoting abusers in order to buy their silence may serve only to encourage future abuses (Straub and Nance, 1990).

Any disciplinary actions taken should be appropriate to the abuse. Possible disciplinary action include reprimands, suspensions, fines or dismissal.

Analysis
User behavior on current systems is analyzed.. Any undesirable behavior that could increase the vulnerability of the planned system is identified and documented.

Design
All elements of the proposed system are analyzed to determine appropriate behavior in regards to that element. Disciplinary actions are determined for each infraction of these guidelines. It is then documented in the user agreement (see Security Policy: User Agreement). Disciplinary action will be fair, appropriate and reflect current legislation. Any historical documentation involving computer abuse within the organization is consulted. This and current legislation are then used as a yardstick to determine appropriate disciplinary actions to be taken in the case of an abuse.

Maintenance
Any modification of legislation or of the computer systems will lead to a review of disciplinary actions. Periodic audits of user behavior should be conducted in order to be assured as to the proper use of the system. Random periodic audits have proven to be an effective deterrent to computer abuse (Hoffer and Straub, 1989).

Classification: Security Policy

Method: Computer Use Access Control Administration

Description: A formal procedure that must be undertaken by any users wishing to gain access, or change their current access, to the computer system.

This procedure should be administered by either a computer user coordinator (see Personnel Practices: Computer User Coordinator) or by the computer security officer (see Personnel Practices: Computer Security Officer).

A special form should be filled out by the requesting user indicating current level of access, access requested and signatures of appropriate managers (Parker, 1982).

In return the user should get a modified user agreement explaining any new rights and responsibilities as well as and disciplinary actions that will be taken if these are not met. The user must sign this before access is granted or privileges are increased (see Security Policy: User Agreement).

Although this adds a level of bureaucracy and complexity to the organization, the advantages gained with proper access control management and user awareness through the user agreement outweighs the negative aspects of this method.

Organization

Method	Description	Cost
Isolation of Sensitive Computer Jobs(Parker, 1982)	Sensitive jobs should be restricted to trusted computer base (TCB) as well as to trusted individuals	
Computer Security Committee (Parker, 1982)	A Committee is formed to establish the computer security policy and to oversee security functions.	
Job Rotation(Ruder and Madden, 1978)	People in data handling jobs should be rotated out of their positions in order to prevent falsification of data. The new person must guarantee the integrity of the data upon entry into the new job.	
Separation of Duties(Clark and Wilson, 1987; Ruder and Madden, 1978)	Separation of duty (or division of responsibility) prevents a single user from defrauding the computer system. Fraud is thus only possible with collusion.	

Classification: Organization

Method: Isolation of Sensitive Computer Jobs

Description: The execution of particularly sensitive jobs, such as year end reporting, or the manipulation of sensitive company or personnel information may require special care. Systems processing personal information need special handling.

Such jobs should be assigned to trusted individuals on a trusted computer base. If this is not possible, then a system could be isolated (by disconnected all communication lines other than those being used and eliminating all but necessary terminals) before the job is processed.

Extraordinary security measures can be taken during the processing of these jobs. Although it may be difficult to schedule the processing of such a job around regular operations, the identification and appropriate execution of these jobs will lead to a drastic reduction in vulnerabilities due to a concentration of security measures (Parker, 1982).

Analysis
The sensitivity of jobs to be executed on the system is evaluated. If the system will be used for the execution of sensitive jobs, then proper consideration of this will be built into the system.

Design
If necessary, tools and procedures used to isolate the system and/or the sensitive job are designed into the system. If the system will only execute sensitive jobs, it will be designed with accordingly high levels of isolation (a trusted computer base of appropriate classification is suggested(CSSC, 1993; DOD, 1983). If the system is used for both sensitive applications as well as regular operations, a method should be found to isolate the job from the rest of the system.

Maintenance
Periodic audits are held to determine the sensitivity requirements of the system. If this is changed in anyway, then appropriate actions will be taken. If the sensitivity levels have decreased, the tools used to promote isolation should be removed in order to remove overhead from the system, as well as to avert the potential disaster of these tools being misused. If the levels have increased, then tools should be purchased or developed to isolate the sensitive jobs from the rest of the system.

Classification: Organization

Method: Computer Security Committee

Description: The computer security Committee is a high-level organization formed of representatives from each part of the organization dealing with the computer system. They are responsible, along with the computer security officer (see Personnel: Computer Security Officer), for the development and application of the computer security policy (see Security Policy).

Their responsibilities include, but are not limited to (Parker, 1982):

1- Coordinating computer security.

2- Periodic reviews of the state of security within the organization.

3- Ensuring the visibility of management's support of the computer security plan.

4- Approving computer security reviews.

5- Receiving and accepting computer security review reports.

6- Ensure proper control interfaces among organizational functions. A proper control interface is essential in assuring that all members of the organization respect the security guidelines set forth by the Committee. This interface can be formed of policies, procedures or any combination thereof.

7- Ensure that privacy and security are part of the overall information handling plan.

Any computer security reviews and recommendations for major controls must be submitted to, and approved by, this Committee.

Although the formation of a computer security Committee will add a level of bureaucracy to the organization, and therefore slow down decision making, the presence of a management organization will demonstrate to all employees management's commitment to a secure environment.

The presence of the Committee, particularly if the formation of the Committee crosses organizational lines, will assure that security is met for all aspects of the computer system, as well as taking into account other management considerations such as employee productivity. Policies and procedures can be more easily enforced with obvious management support.

Computer security, particularly in a commercial environment, should take a more organizational approach, rather than a technical one. The presence of a Committee will help to achieve this goal. Managers not on the Committee must be made aware of their responsibilities for the security of their department. (Parker, 1982)

Analysis
The computer security Committee is formed. If it already exists, it may be modified to include management that will be affected by the development of the new system. All elements of the security plan must be presented to the Committee for approval before the systems is designed. The Committee must assure that any security and organizational requirements are met by the new system.

Design
Once the design of the system is complete, the Committee will meet with the analysts to assure that both organizational and security concerns are met.

Implementation
The Committee must demonstrate their dedication to secure organizational operations by their presence and any security briefings, or educational meetings, to which their employees are subjected to. Employees must be made aware of management's support of security policies and practices from the birth and introduction of the new system, if these policies and practices are to be considered legitimate.

Maintenance
Any significant changes to security policies and practices must be reviewed, accepted or rejected by the Committee. The Committee does not have to manage day-to-day security operations (see Personnel: Computer Security Officer) but any major changes to security practices must meet the approval of the Committee. In this way the organization can be assured that not only are they operating in a secure environment, but also an organizationally effective one. The Committee will also review any audit reports generated by the organization. Any problems identified by the auditors should be immediately addressed by the Committee to assure the maintenance of a secure work environment.

Classification: Organization

Method: Job Rotation

Description: Job rotation reflects those policies and procedures to periodically rotate those positions that have a great deal of authority among individuals in the data handling process (Ruder and Madden, 1978). For example, the person responsible for entering accounts information in a bank should be replaced by a new person periodically and without notice. The new person's first task is to assure the integrity of the data.

This policy also greatly increases the effect of the separation of duty tool (see Organization: Separation of Duty). With random (or seemingly so) job rotation, and proper Separation of Duty, collusion will only be possible by chance.

The schedule is developed by the design team in cooperation with the computer security Committee. This assures that not only computer security is being followed, but that organizational concerns are also being met.

The job rotation schedule should be confidential to minimize the risk of collusion. Only the computer security Committee or the computer security officer should have access to the list.

Analysis
Jobs which have a great deal of importance in the data handling process are identified and fitted to the Separation of duty schedule (see Organization: Separation of Duty). All employee qualified to perform each critical job are identified.

Design
Job rotation schedules are designed along with the system. They not only reflect the pool of possible employees, but reflect proper Separation of duty rules.

Implementation
Employees are educated as to the responsibilities of each role they may assume. They are trained in how to properly validate the data they are responsible for upon acceptance of a new role. They are also informed of the methods and reasoning for the random job rotation schedule.

Maintenance
Any new employee is fit into the job rotation schedule. They are educated as to their role in Separation of duty.

Classification: Organization

Method: Separation of Duty

Description: Separation of Duty (or division of responsibility) breaks down all EDP functions into as many discrete self-contained activities as is practical and cost-effective under the circumstances. Accountability issues are also broken down into discrete areas to facilitate assignment of responsibilities (Parker, 1982). The result is a reduction in the level of trust needed for each manager by reducing the possibility of accidental or intentional acts resulting in loss. It forces the need for collusion among individuals who may attempt unauthorized activities.

When separating jobs into discrete activities, care should be taken to ensure external consistency of the data objects. Functions must be broken down, but data objects should not be. Activities resulting from Separation of duty should, at their lowest level, represent the data object they deal with.

By separating jobs into separate activities, and assuring that these activities can only be completed with well-formed transactions (see Application Security: Well Formed Transaction), only collusion between employees can result in fraud. Coupling Separation of duty with random job rotation (see Organization: Job Rotation) will make collusion very difficult, thus assuring the integrity of data. If collusion is occurring, it should be discovered quickly when an employee accepts a new role and validates their data.

To ensure the maximum protection, Separation of duty must be coupled with the concept of well-formed transactions. A well formed transaction is one which contains all necessary validation elements and attempts to accomplish only what it was created for. It is created in most situations, not by a user, but by a process. In this way, each transaction is discrete and formed by a different individual executing a process. If each step is performed by a different person, the external and internal representation should correspond unless some of the people conspire (Clark and Wilson, 1987).

The first rule of Separation of duty is that any person permitted to create or certify a well-formed transaction may not be permitted to execute it. This rule ensure that at least two people are required to cause a change in the set of well-formed transactions (Clark and Wilson, 1987). For example, any in-house application programmers who are permitted to create processes containing legitimate, well-formed transactions, must not be allowed to execute these application on legitimate data. By the same token, those who may execute applications must not be allowed to modify them.

The easiest way to enforce Separation of duty rules is to separate applications into functional groups by job type. Each employee will then gain permission to execute a subset of programs, based on their job function. In this way, system security elements will be used to enforce the Separation of duty criterion (see System Security: Least Privilege Principal).

It is important to note that Separation of Duty and well-formed transactions are effective only if users are not allowed to directly modify the data. In a military environment, a data item often has a security level associated with it. In a commercial environment, it is often much more effective to link a data item with a set of applications that have permission to read or modify it. In this way only valid transactions will be performed on the data item. In addition, a user is not given permission to modify data, but rather the permission to execute a process. These two differences are critical in making Separation of duty and well-formed transactions effective in controlling error and fraud.

Separating jobs into discrete, self-contained units helps to create more efficient EDP functions. However, some smaller organizations may find it difficult to separate functions sufficiently. If the organization is so small that it has difficulty separating tasks into discrete functions, it is probably unnecessary to do so. Separation of duty is necessary only when there is the possiblity of unnoticed modification of applications or data. If this is not the case in an organization, then separation of duty can add a great deal of complexity unnecessarily.

Analysis
Job functions are separated into discrete groups. The computer security Committee (see Organization Computer Security Committee) must be involved in the separation of duty because of the obvious organizational impact that such a task will have. It is important to isolate activities without significantly reducing employee productivity. This can be accomplished by breaking down each main activity into its components. For example, if the activity is purchasing, it may be broken down into placing the order to purchasing, purchasing putting a request in for funds to the accounting department, the accounting department confirms the expenditure, the order is then placed with the supplier by the purchasing department, the check is issued to the supplier by the accounting department, the item is received in receiving and documented, and accounting documents the received check. In this manner, if each of these function is composed of a well formed transaction and performed by different people (eg. 3 people in accounting, one to approve the purchase, one to issue the check and one to document the receipt of the cancelled check) then the possibility of fraud is greatly reduced. All persons involved would have to be in collusion for fraud to be possible in such a situation. Note however that this does not mean that there is one person in accounting whose only job is to document received checks. A person may be responsible for many different areas, and many wellformed transactions.

Design
The system is designed to accommodate the job functions. Applications are divided into subsets by job function. Execution permissions are set in such a way as to restrict their use to valid users.

Implementation
Users are informed of the responsibilities and, if necessary, their new roles within the organizations. The job rotation schedule is explained. The reasoning behind well formed transactions, division of responsibility and job rotation is explained to the user.

Maintenance
New users are given orientations sessions. The system may have to modified to take into account new roles and functions. Periodic audits are conducted to assure that the division of responsibility is working as intended.

Physical and Fire

Method	Description	Cost
Physical Security Perimeter and Access Barriers (Parker, 1982; Ruder and Madden, 1978)	Preventing access to the physical boundary of security importance.	
Personnel Authentication (Parker, 1982)	Physical means of identifying personnel.	
Backups and Offsite Storage (Ruder and Madden, 1978)	Conducting regularly scheduled backups of all important data in the organization. Offsite storage of backups in case of disaster.	
Electrical Power Shutdown, Recovery and Safeguards (Parker, 1982)	Policies and procedures for protecting sensitive electronic equipment from electrical accidents. Includes procedures for emergency shutdown and recovery.	
Fire Detection and Prevention (Ruder and Madden, 1978)	Precautions for fire prevention, detection and extinguishment.	
Computer Inventory Control (Ruder and Madden, 1978)	Inventory control of all parts and equipment, including location, useful life, date of purchase, date of installation.	
Alternate Communication Paths (Ruder and Madden, 1978)	Ensuring the existence of alternate communication paths to critical online systems.	
Computer Terminal Access and Use Restrictions (Parker, 1982)	Restrict use or access to terminals to certain rooms or during certain times.	

Classification: Physical and Fire

Method: Physical Security Perimeter and Access Barriers

Description: A physical perimeter must be defined and clearly marked. This perimeter should contain all critical equipment including, but not limited to: computers, terminals, peripherals (such as printers, modems), network junction boxes, electrical power switching, telephone junction boxes and fire extinguishing equipment.

A security perimeter must be easily identified to avoid accidental intrusion. It should be easily discernible, simple, uncluttered and sufficiently secure relative to the value of the assets contained within the perimeter. Drawings and specifications of the perimeter must be available and used for planning any facility changes. (Parker, 1982) Areas above and below (particularly false floors) should be considered when securing the Perimeter.

Perimeters within a perimeter are permitted, and often serve to simplify the layout. A perimeter may contain a second (third, fourth...) perimeter of higher security level.

The establishment of perimeters helps to ensure maximum protection of all critical facilities as well as allowing facilities to be modified without compromising security. However, a perimeter that is too obvious to the public may attract unwanted attention. (Parker, 1982)

Physical Access Barriers

Physical access barriers are used to protect perimeters. These include, but are not limited to: strong materials between perimeters, sign in/out log, challenge of authorized persons by unauthorized persons (this should be an employee responsibility, and should be included in the user agreement (see Security Policy: User Agreement), posted signs explaining which areas are restricted, mechanically or electronically locked doors, guards (local or remote using security cameras), mantraps or turnstiles, internal tampering alarms (including alarms against unplugging equipment), metal detectors, xrays, and package controls. These barriers may not be appropriate for all commercial environments, and should be implemented as necessary. The barriers implemented should reflect the value of the information/equipment being guarded. Operations area surveillance, either with local guards or closed circuit television can be one of the most effective computer abuse detection tools available. (Ruder and Madden, 1978)

All work environments should have appropriate barriers, with the possible exception of public entry lobbies, lavatories, lounges, food areas and all areas outside the outermost security perimeters. (Parker, 1982)

Access to secure perimeters should be administrated by a central reception area. This area should have access to an authorized access list for each perimeter. Employees and in-house vendors should be allowed access to perimeters on a least-privilege basis.

All procedures involving perimeters should be well documented, this includes exception procedures.

The implementation of appropriate perimeters and barriers helps maintain a security awareness among employees and helps discourage malicious acts. It is important, however, that the stringency of controls match actual needs, or employee productivity may be affected. Any barriers that can be automated should be.

Analysis
Any documentation of security perimeters should be analyzed. Necessary security precautions for the new system should be listed. Perimeters should be evaluated to determine their appropriateness for the new system.

Design
If existing perimeters are insufficient for the new system, existing perimeters should be enlarged, new access barriers should be added or a completely new perimeter should be designed to house the system. It is important at this stage to determine the geographical location of the new system. Fire detection and prevention equipment, access barriers and any other job which may require modification to the building must be completed before the new system is implemented.

Implementation
The system is housed in its new area only when all security precautions have been completed. The perimeter must be tested once the system is implemented to ensure a secure environment before employees are allowed access to the system.

Maintenance
Any changes to the system, or any security breach, should lead to an audit into the effectiveness of the security perimeter and all barriers. Periodic audits of the perimeter should be conducted to ensure that all barriers are functioning correctly and that all users are behaving within their responsibilities. New employees should be made aware of the security perimeters that they have access to and what their responsibilities are in maintaining a secure area.

Classification: Physical and Fire

Method: Personnel Authentication

Description: This method is concerned with the identification of legitimate personnel and their perimeter authorization.

Determining who is authorized to enter a physical security perimeter can be accomplished through the use of color coded badges with photographs. Different colors are used to differentiate between employees, vendors, temporary badges etc.

The decision to require badges depends on business practices, number of people, amount of traffic and other access controls in use (Parker, 1982).

Badges may be required as identification only in certain perimeters. There are certain advantages to badges. They allow for a quick visual inspection of the authorization of an individual. They permit regular employees to enforce perimeter security restrictions by being able to quickly and efficiently identify unauthorized personnel. Their use also serves to dissuade any attempts at unauthorized entry of the perimeter.

Separation of duties (see Organization: Separation of Duties) is also strengthened by the use of identification badges as collusion may not be possible if identification badges restrict movement through the company.

The use of identification badges must be policed if it is to provide proper security. New technology can be taken advantage of by allowing security badges to double as keys for magnetic card locks. This not only helps for identification but also automates the movement restrictions between perimeters.

Badges are not for use by every organization as they require a fair amount of overhead. Also, if the geographical boundaries of the organization are limited (to one room, for example) then the use of badges is strictly unnecessary.

Badges are used to enforce perimeter rules, and the systems development life cycle dictated in Physical: Physical Security Perimeter and Access Barriers should be followed, with the necessity of badges being taken into account when determining perimeter restrictions.

Classification: Physical and Fire

Method: Backups and Offsite storage

Description: Periodic backup schedules are critical to proper recovery in case of a disaster, or simply to reload an accidentally corrupted or deleted file.

Daily backups of regularly modified data files or applications will assure that minimum work is lost in case of an unforseen event. Backup schedules should be designed to allow flexibility of recovery. Backups should be available from up to year previously.

Offsite backups should be kept to allow recovery in case of a fire or natural disaster. Media should be verified prior to backups to assure it's integrity. Backups are useless if they are destroyed in the disaster or kept on media that does not allow proper recovery.

Procedures are defined for reloading files or backups. User should be made aware of what procedure to follow should data from a backup be needed.

Analysis
A backup schedule is designed. This schedule will allow for regular backups, a minimum use of media and a time span that will guarantee a minimum loss of work involved.

Implementation
All applications and data files are backed up prior to system implementation.

Maintenance
Regular backups are made. Their integrity is verified before they are sent offsite or put into local storage.

Classification: Physical and Fire

Method: Electrical Power Shutdown, Recovery and Safeguards

Description: This tool is critical to all computer sites. Power failures are relatively frequent in all parts of the world and, in most case, are unexpected. Each piece of equipment that is separately powered should have it's own circuit breaker. Special emphasis should be placed on critical systems, particularly those that contain CPUs or critical data.

Equipment should also be equipped with surge suppressors, power cleaners or some other device to protect against force spikes.

Circuit breakers should be clearly marked for manual activation. The location of all breakers should be documented in the disaster recovery plan (see Security Policy: Disaster Recovery).

Master circuit breakers (circuit breakers which control power to all equipment) should be located next to each emergency door. They should be clearly marked and include instructions for use. This is particularly useful for emergency response personnel such as firefighters or rescue workers.

Documentation must be created to explain all procedures that must be followed in the case of an emergency power off. This document must include a list of personnel

responsible for bringing the system back online, as well as procedures for each machine detailing steps that must be followed when lost data from backups.

The decrease in price of uninterruptible power supplies (UPS) or alternate power supplies (APS) has allowed all systems their benefits. Such a system may be installed on specific machines, or a more powerful version can be employed to allow all equipment a few minutes of continued operation after a power failure, minimizing the loss of data and simplifying the recovery procedure. If several of these devices are used, and their targets overlap, then they should be stacked differently to reduce the chance of both failing.

Periodic testing of circuit breakers, UPS, APS and recovery procedures will increase assurance in the system.

Breakers should be located in a secure perimeter to reduce the chances of tampering. Accidental or malicious use of breakers could be very harmful.

Analysis
The risk analysis should be able to isolate potential electrical risk associated with a particular geographical area. The system should be designed with theses risks in mind. Current location of circuit breakers should be considered when determining the geographical location of the new system.

Design
Any modifications needed to the structure that will house the system should be implemented before the system is installed. Circuit breakers for each main component of the system should be available as well as a main circuit breaker located adjacent to an emergency exist. The design of the system will include an APS or UPS, as is necessary. This equipment can target specific elements of the computer system, the entire system or the entire company.

Implementation
Breakers, UPSs and APSs should be thoroughly tested before the system becomes operational. Users should be instructed on the use of breakers, and what procedures to follow in case of a power failure, or power shutdown.

Maintenance
Periodic tests are conducted to ensure the effectiveness of recovery plans. UPS and APS systems are also periodically verified to assure their usefulness in case of an emergency.

Classification: Physical and Fire

Method: Fire Detection and Prevention

Description:

Smoke and fire are a critical threat to computer installations. Fire was the leading cause of computer room losses in the 1980's, resulting in over $11 million of damage.

Precautions can be taken to alleviate the risk associated with fire and smoke. The first step is to ensure that the building housing the computer facilities is constructed to be as flame retardant as possible. This includes non-combustible building materials, such as fire-rated wired glass for exterior windows and doors. Doors should have a minimum 1 hour fire rating. (Factory Mutual System, 1993) Large or particularly valuable sites should have as many smoke tight subdivisions as possible. Air ducts should be fitted with smoke alarms if incoming, smoke tight dampeners if outgoing.

Programmable high-sensitivity smoke detectors or beam-type detectors should be present in all computer facilities, and should protect all areas including areas under raised floors and above suspended ceilings. Ideally, an engineering survey should be performed on the area to determine the best placement for smoke detectors and fire extinguishers. (Factory Mutual System, 1993)

Upon activation of the first smoke detector or first level alarm (Factory Mutual System, 1993):

- an alarm will sound locally in the computer centre, to inform those responsible of the area and nature of the concern.
- the ventilation system smoke control is initiated.
- initiate manually controlled shutdown of computer units that appear to be the source of the smoke.
- pre-action sprinkler system valves are tripped, allowing water into the pipework.
- Personnel are evacuated from any sites with second-level alarms linked to gaseous extinguishing systems.

Upon actuation of second-level alarms:

- alarm is transmitted to a control station or the fire department.
- all electrical power to computer systems is shut down.
- all dampeners on ventilation ducts leading in or out of rooms with gaseous extinguishment systems are shut and the extinguishers should be discharged.

Note: Halon as a fire extinguishment agent was banned September 16th, 1987 due to the damage it causes to the ozone layer. FM-200, a clean-air fire suppression system, is now available as an alternative to Halon 1301.

Analysis
The physical facilities should be reviewed. An engineering survey should be performed on the area to determine the best placement for smoke detectors and fire extinguishers. All areas should be covered, including areas above suspended ceilings and below raised floors. Communication paths for the fire alarms should be carefully detailed. An emergency system shut down plan should be designed to allow the safe shutdown of critical systems at the sounding of a first level alarm.

Design
All alarms, fire extinguishment systems, ventilation ducts with dampeners and other precautions should be installed and tested prior to the installation of the computer system

Implementation
The computer system is physically installed. Manual shutdown switches are clearly labelled. The computer system is placed in close proximity to smoke alarms to maximize their effectiveness.

Maintenance
All smoke detectors, fire extinguishment systems and manual shutdown plans are periodically tested to ensure their effectiveness. If the facilities are expanded, new areas should be included in the fire security perimeter. This includes the installation of new smoke detectors and fire extinguishment units, and the modification of current fire plans to encompass all new areas.

Classification: Physical and Fire

Method: Computer Inventory Control

Description: Procedure and software are developed to keep tract of inventory of all computer equipment. This includes but is not limited to: hardware, hardware replacement parts, used and unused media, and supplies such as paper, ribbons, cartridges from their arrival until the end of their useful life.

When equipment is allocated, a particular individual should be given complete responsibility for it. He/she should be aware of it's proper maintenance and the procedures to follow should it fall into disuse/disrepair or is stolen.

This list/program will then be used by the auditor to ensure the presence and good working order of all equipment in the company.

Analysis
If this list does not already exist, it is created for all equipment currently in possession of the company. This list can be used to determine what assets are available to the new computer system and what must be purchased.

Design
All equipment used in the new system must be assigned to an individual who will be responsible for its availability and its proper working order.

Maintenance
Periodic equipment audits are conducted to assure the presence and functionality of all equipment. The list is modified with each new purchase or change of equipment responsibility.

Classification: Physical and Fire

Method: Alternate Communication Paths

Description: This is a precaution to prevent denial of critical services.

Hardware and facilities are used to provide alternate communication paths to critical online systems. Alternate lines to the central office of the phone company, or direct lines to the external facility can be obtained. The cost of leased lines is reasonable and may be worth it to maintain services to critical external sites.

Analysis
The importance of communications between sites is ascertained.

Design
If communication paths are critical, the system will designed with alternate communication paths.

Implementation
The alternate communication paths are tested to assure their validity before the system goes online.

Maintenance
Periodic testing of communication paths is conducted to assure their reliability.

Classification: Physical and Fire

Method: Computer Terminal Access and Use Restrictions

Description: The purpose of this method is to restrict access to terminals to authorized users. All terminals should be located within a secure perimeter to minimize potential misuse. Automatic terminal timed shutoff should be implemented to restrict unauthorized access to an unoccupied and forgotten terminal.

Time cards can be used in high security level environments. These cards are placed in a card reader which authorizes computer use by monitoring the card. When the card is removed, the terminal is automatically shut down. Some cards contain a small processor which generates random passwords authenticated by the system. These cards can be used not only to monitor terminal use, but also as user authentication (see System Security: User Authentication).

A procedure should also be put into place which users must undertake if they wish access granted or modified to certain terminals. This procedure should be administered either by the Computer User Coordinator (see Personnel Practices: Computer User Coordinator) or by the Computer Security Officer (see Personnel Practices: Computer Security Officer). The procedure should include a form listing current access privileges, requested access privileges and authorization of appropriate managers in the form of a signature.

A document explaining the responsibilities accompanying the new access privileges should be given to the user. A new user agreement may have to be signed based on the new privileges (see Security Policy: User Agreement).

These policies and procedures will add complexity to the physical system, as well as bureaucracy to the organization. Their use, however, can be a very effective tool in managing security issues and maintaining access controls within the organization.

Managers should be informed that access to terminals and perimeters should be granted on a least-privilege basis.

Design
The security level of the system is assessed, and appropriate measures (time cards, automatic shutoff, placement of terminal in appropriately secure perimeters) are taken to assure that only authorized user have access to the terminals. Procedures are put into place to allow users to modify their access if necessary.

Maintenance
Periodic assessments of terminal security levels are conducted to ensure that security levels assigned to terminals are still appropriate. Audits are conducted to ensure that security measures have not been bypassed and that users have access to terminals on a least privilege basis.

Personnel Practices

Method	Description	Cost
Computer Security Officer (Hoffer and Straub, 1989; Wong, 1986; Parker, 1982)	A CSO should be established and trained to oversee all computer security activities in the organization.	
EDP Auditor (Parker, 1982)	Personnel should be properly trained to conduct audits on the organizations computer system.	
Computer User Trouble Calls Logging (Parker, 1982)	All reported problems should be logged to prevent overlooked security problems.	
Cooperation of CSOs (Parker, 1982)	Computer security officers should cooperate with CSOs from other organizations to exchange information.	

Classification: Personnel Practices

Method: Computer Security Officer

Description: A computer security officer (CSO) is responsible for overseeing all tasks relating to computers security within the organization. It is his/her responsibility to assure proper security levels are met in the data centre, communications system, network, terminals, and personal computers throughout the organization. (Wong, 1986)

Among the CSOs other responsibilities are (Wong, 1986):

- Liaison with computer users.
- Preparation and enforcement of security standards and procedures.
- Assure that the organizational policies comply with legislation, such as the data protection act.
- Risk analysis and monitoring day-to-day events to make sure that controls are sufficient. This task may be the CSOs sole responsibility, or he/she may share it with the EDP auditor.
- Implementation and administration of access control equipment and software, and control of encryption and authentication devices.
- Contingency planning and procurement of computer insurance.
- Analyze and advise on new legislation.
- Evaluate cost-effectiveness of controls.
- Motivation of management and user to proper use of controls.
- Managing security policy and disaster recovery plan.

- Review user trouble call logs.
- Keep abreast of new technological and security developments.
- Act as liaison between different functional areas regarding computer security behavior.
- Conduct random-schedule security reviews.
- Review user behavior to assure that the end-user agreement is being upheld.
- In smaller organizations, the CSO may have other tasks above and beyond his/her regular responsibilities.

The CSO must be able to communicate at all levels to be effective. He/she should understand corporate security requirements at all levels and recommend, cost-justify, and implement the necessary safeguard in strategic areas.

The CSO must keep him/herself informed off all corporate plans and the future strategy governing computing and communications including: new data centres, use of encryption on communication lines, computer acquisition, new projects and diversification or consolidation of new product lines. (Wong, 1986)

In order to remain effective, the CSO should report directly to the chief executive officer or to the board of directors. This will allow him/her enough authority to

effectively implement security plans, and to discipline those in the organization who act against corporate security.

Use of outside expertise should not be discounted, particularly in those areas where there is a shortage of in-house expertise. (Wong, 1986)

The appointment of a CSO provides a focus for the formal development of a computer security program. However, the CSO must not be viewed as a scapegoat by others in the organization. Managers in particular should be aware that, despite the presence of a CSO, they are still responsible for enforcing security rules within their departments. (Parker, 1982)

The appointment of a CSO within an organization transcends the SDLC. He/she should be appointed to govern security within an organization as soon as possible. Preferable before any system is designed. Risk analysis and other tasks should be accomplished before the organization implements any computerized systems. If computerized systems exist and the organization does not have a CSO, one should be appointed at the earliest possible time and should be allowed to familiarize him/herself with corporate security policies before any other systems are developed.

Classification: Personnel Practices

Method: Electronic Data Processing (EDP) Auditor

Description: An EDP auditor is only necessary in larger organization, where the CSO cannot manage all the responsibilities normally attributed to that position.

The EDP auditor's tasks include:

- monitoring the effectiveness of security policies procedures, software and training. (Hoffer and Straub, 1989)
- developing an audit plan in conjunction with the CSO. The audit plan should target area with the highest risk first, to assure acceptable security levels in the most vulnerable parts of the organization. (Parker, 1982)
- make sure that audit logs are complete, meaningful, accurate and tamper proof. (Wong, 1986)
- perform analysis on logs to ascertain whether or not control requirements have been met. (Wong, 1986)
- analyze any problems relating to adequacy in staffing
- determine the cost effectiveness of controls
- evaluate methods by which controls can be circumvented
- determine how actual administrative and running costs compare with budgeted expenditures
- analyze whether or not control requirements have been adequately met
- evaluate the effect of controls on employee productivity

Periodic review of recorded incidents should provide constructive comments on user experience, attitudes and utilization of control procedures, software and devices. Should also provide details on shortcomings or benefits created by applying such controls. (Wong, 1986)

Analysis
Prior to the development of any new system, a CSO or EDP auditor is appointed (depending on organizational needs). An audit of all existing systems and business practices is conducted, along with a formal risk analysis. Findings are analyzed to determine the greatest areas of risks in the organization. The system is the developed with controls to offset these risks.

Design
The auditor is responsible for making sure that controls are designed into the system. It is the auditor's responsibility to assure that the new system's risk levels fall within the range of acceptable risk.

Implementation
The auditor should be present to assure that the controls have been implemented correctly and that the users are familiar with their reasoning as well as their use.

Maintenance
Most of the tasks completed in the maintenance stage are conducted by either the CSO or the EDP auditor. The EDP auditor's main function is the constant monitoring of all the systems in the organization to assure that controls are still effective. Risk should be periodically evaluated to maintain assurance as to acceptable risk levels within the organization.

Classification: Personnel Practices

Method: Computer User Trouble Calls Logging

Description: User problem call are logged by operation staff. Caller's name, date, time and nature of the call should be logged for historical reference. A brief description of each call, and what solutions were applied by operations staff is prepared for review by the organization's CSO. The CSO then determines what new areas of risk the company is exposed to, and whether current controls are adequate. Any actions taken by operations staff are also evaluated to determine whether they suffice as problem resolutions, whether or not any adverse effects are generated by the solution and, finally, whether or not the operations personnel had the authority to implement the solution. (Parker, 1982)

The advantages to such as system is that it forces users and staff to justify any actions taken regarding the controls of the system. The documentation generated by such a system is also very useful in determining adequacy of controls and to determine what controls will be necessary in any new system implemented.

Any impact that controls have on performance will also be reflected in the user logs. Despite the red tape generated by such a system, the advantage that user trouble call logging confers to the management of current and future controls justifies their use.

Analysis
Past trouble call logs are reviewed to determine what risks the new system will be facing, what controls are no longer necessary and which new controls should be added to keep risk with the accepted range.

Maintenance
Trouble call are regularly reviewed by the CSO and/or the EDP auditor in order to evaluate the ongoing effectiveness of implemented controls.

Classification: Personnel Practices

Method: Cooperation of CSOs

Description: Computer security organizations already exist in most major cities. Security personnel from many private and public sector companies gather at regular meetings to exchange information that will advantage the group. In this manner, mistakes are not repeated. The CSO from one organization can seek advice from others who may have already experienced and found solutions to the problem. (Parker, 1982)

Emergency "hot-lines" can be formed to disseminate information on an emergency basis. National and international organizations can also be used to glean important information about current computer security risks. CERT, the Computer Emergency Response Team is just such an organization.

It is important that the information shared with such groups not weaken the organization to which the information pertains. Classified information should remain classified. CSOs involved in such security groups should share information with discretion and assure that they have not weakened the security position of their organization. (Parker, 1982)

Analysis
Any historical information that can be obtained from these organizations and that pertain to the system being analyzed should be reviewed. Risk analysis can also be greatly facilitated by reviewing past risk analysis' done by others in the same industry and in similar geographical boundaries.

Maintenance
Such organizations should be monitored to determine what new risks, if any, have emerged. If risks are identified, the security personnel from all organizations affected by the risks can work together to determine what controls can be used to offset the risk.

Insurance

Method	Description	Cost
Financial Loss Contingency and Recovery Funding (Parker, 1982)	Insurance in case of business. Emergency funds in the case of self-insured agencies.	
Contingency Recovery and Replacement (Parker, 1982)	Alternative source of equipment in case of failure (emergency replacement policies from vendors).	

Classification: Insurance

Method: Financial Loss Contingency and Recovery Funding

Description: Insurance or emergency funds (for self-insured organizations) should be available in case or contingencies or recovery. Specialized computer insurance should be purchased for areas in which regular insurance does not apply.

This insurance should cover:

- asset losses
- business interruption
- extra expenses resulting from contingency recovery

Organization that are not self-insured should bond all employees in high-risk EDP activities against fraud. A blanket bond is usually sufficient for this.

It is important that insurance be considered an addition, and not a replacement, to proper security practices. (Parker, 1982)

Analysis
Any new system must be evaluated to determine whether or not it, and the processes it controls, are covered under current corporate insurance policies. If current policies or emergency funds are insufficient, actions should be take to assure that the new system is well protected financial in case of contingency or recovery.

Maintenance
Periodic monitoring should be used to assure that current contingency funding or insurance is till sufficient to cover the system in case of disaster.

Classification: Insurance

Method: Contingency Recovery and Replacement

Description: The disaster recovery plan (see Security Policy: Disaster Recovery Plan) should include a written vendor commitment to replace critical equipment and supplies in a specified period of time following a disaster or other event leading to equipment loss. This document should be legally valid, and specify all terms of the replacement including the equipment to be replaced, the period of time that can elapse before replacement and any financial renumeration required.

The disaster recovery plan should outline all steps that must be taken in order to regain critical services. Installation procedures should be well documented, unless the vendor agreement specifies that it is the vendor's responsibility to install the new equipment. In some cases a promise of best effort from the vendor may be all that you can expect.

Contingency equipment replacement provides the organization with a means of planning alternative data processing until equipment and computing resources have been restored (Parker, 1982).

Other contingency precautions are possible. These include providing "hot sites", or office space with alternative computer equipment to be used in case of disaster. "Warm sites" or leased sites are those which are leased from outside sources such as hotels or

computer companies, and include computer equipment necessary to continue operations.

Analysis
How critical is the new system to organizational operation? How long can the organization operate without the system? These and similar questions are answered at this stage in order to determine the role and importance of the new system.
Design
If it has been ascertained that the system is critical to the organization, precautions must be taken to ensure its continued operation. A written vendor commitment is obtained to have the equipment replaced in a reasonable time. If the system is so critical that no downtime is acceptable, a hot site or warm site should be arranged to assure a continuation of services.
Implementation
If a hot or warm site is necessary, it will be implemented in parallel with the system under design.
Maintenance
Vendor contracts are renewed periodically to assure their completeness and correctness, this is particularly important if the system undergoes periodic hardware changes. Hot sites and warm sites should also follow system upgrades to assure their role in disaster recovery.

System Security

Method	Description	Cost/Eff.[1]
Firewalls(Ranum, 1994)	Dedicated hardware and software used to give access to networks, particularly the Internet, while reducing an organization's vulnerability to external threats	0.9444
Remote Terminal Physical Security (Jamieson and Low, 1989; Parker, 1982)	Assuring that remote terminals are valid and secure	0.8493
Restricted Use of System Utility Programs (Parker, 1982; Ruder and Madden, 1978)	Allowing access to system utility programs only to authorized users.	0.5638
Assign File and Programs to Users (Parker, 1982)	Responsibility for the availability and maintenance of files and programs should be attributed to the user, when appropriate.	0.7883
Data Classification (Parker, 1982; Ruder and Madden, 1978)	Access labels should be attached to each piece of data (file, record, field) depending on the granularity of the operating or database management system.	0.7783
Bell-Lapadula Data Rule[2] (McHugh and Thuraisingham, 1988)	Use of the simple and *-properties when determining read or write access to objects	0.8
Technical Review of Operating Systems Changes (Parker, 1982)	Each change to the OS should lead to a detailed technical review of the changes, and their impact to the organization	1.0678
Cryptographic Protection (Parker, 1982; Ruder and Madden, 1978)	Cryptography to assure data security on local machines and communication lines.	1.1111

Method	Description	Cost/Eff.[1]
User Authentication (Boockholdt, 1989; Wong, 1987; Chalmers, 1986; Parker, 1982; Ruder and Madden, 1978)	Assuring the accuracy of user identification for proper security management.	0.5941
Automatic, Timed Terminal Logoff (Wong, 1987; Ruder and Madden, 1978)	Automatic terminal log-offs after a certain period of inactivity.	0.5091
Bill Back System (Ruder and Madden, 1978)	Billing back computer resource expenses to users. Helps identify unusual resource use.	1.2384
Hardware Monitors (Ruder and Madden, 1978)	Monitoring hardware usage throughout the organization for unusual levels.	0.9592

[1]See page 216 for a description of the cost/effectiveness ratio.
[2]The Bell-Lapadula Data Rule will not be included in the final draft of the framework

Classification: System Security

Method: Firewalls

Description: A firewall is a set of hardware and software used to protect one network from another untrusted network. The typical firewall can be though of as a pair of mechanisms: one which exists to block traffic and the other which exists to permit traffic. Emphasis can be placed on either of these functions. (Ranum, 1994)

Firewalls are particularly important in organizations which wish to have access to the Internet without having to incur the vulnerabilities inherent in this international network. Safety, particularly when connecting to the Internet, is often of paramount importance. A firewall allows an organization to restrict traffic both to and from the untrusted network.

Firewalls can also be used to store information that the organization wishes to make public. In this way information can be disseminated to the public without putting other, more sensitive, information at risk. (Ranum, 1994)

Services that are often blocked by firewalls are: TFTP, NFS, SUNRPC, login, shell, route, syslog, uucp, openwindows, X11 and others. In effect, a proper firewall should block **all** services other than those specified.

Firewalls can often be difficult to implement while maintaining enough openness to allow reasonable freedom in accessing the untrusted network. It is important that the firewall allow enough access to make connecting to the untrusted network useful. If an organization does not allow meaningful operations, or if a machine contains very sensitive data, then the connection to the untrusted network should probably not be attempted. Firewalls are a very good protection from an untrusted network, but they are not foolproof. Every access that is allowed could be used against the organization, to gain access to services that were thought inaccessible. (Ranum, 1994)

Many products are available to help implement a secure firewall. The JANUS Internet Firewall server and SOCKS are two products/tools that should be evaluated for use by the organization. The first product, the Janus Firewall Server provides transparent access for internal users, while blocking any unauthorized access attempt in from the external network. There are many similar products, these can be use to simplify the design and implementation of a firewall. (Dektronix Inc, 1995)

The second SOCKS, is a package that allows hosts behind a firewall to gain full access to the Internet without requiring direct IP reachability. It works by redirecting requests to talk to Internet sites to a server, who authorizes the connection and passes data back and forth. (Kuris, 1994)

Analysis
If the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered. Any operations that need to be available between the system (or network) and the untrusted network is determined, and a firewall is designed to allow these operations. The sensitivity of the system is evaluated, and if the system is deemed very sensitive to the organization, then the network connection is not attempted.

Design
If a network connection is deemed necessary, then the firewall is designed prior to the system. In this way, the system can be implemented behind the firewall when it is ready, and the organization will not be exposed to any unnecessary security risks.

Implementation
The system under design is connected to the firewall, and the firewall is evaluated to assure that it provides appropriate security to the system. If the system remains vulnerable, then it should be disconnected from the untrusted network immediately.

Maintenance
The firewall must be audited regularly to ensure that it is still assuring required security levels between the untrusted network and the organizations system.

Classification: System Security

Method: Remote Terminal Physical Security

Description: Personal computers are often used to access corporate system. They are very rarely under the direct control of organizations operations staff, therefore security on these machines should be of paramount importance. This method insists that computer operations staff act as intermediaries for users accessing the organization's computer facilities from external terminals. In essence, the operations staff must be able to disallow access if proper remote terminal security is not in place. (Parker, 1982)

Signed agreements are used to enforce these security requirements. Users must be aware of what behavior is expected of them, and what behavior will not be tolerated.(Parker, 1982) In addition, it is important that only access from a machine that **must** access the system be tolerated. The vulnerability of such systems can place organizational computer security in considerable jeopardy. (Jamieson and Low, 1989)

If possible, encryption should be used to moderate external export of data. Dedicated links are even better as they will restrict the ability of outside personnel "spoofing" legitimate users of the system. If dedicated links are not possible then a callback system should be implemented.

A callback system does not allow users to log directly in the organization's computer system. Rather, they call the system, which then disconnects the user and calls the user

back at some pre-determined phone number. This will usually restrict the user to some predefined physical location. In order for callback systems to be effective, cellular phone access should be discouraged.

External links should be connected to different network devices to ensure continuity of services should one of the devices fail. In addition, different routing of external links and alternate communication paths will lessen the risk of disruption of service due to cable breakage, industrial action or equipment failure (see Physical Security: Alternate Communication Paths). (Jamieson and Low, 1989)

Other considerations include (Jamieson and Low, 1989):

- digital communication has a lower error rate than analog
- modems with automatic equalization also work to reduce errors
- error detection and correction tools should be used whenever possible to ensure the validity of transmitted data

Analysis
The need for access from external sources should be evaluated. If it is not necessary, or if the data on the system is of a sufficiently sensitive nature, the access from external sources should be disallowed.

Design
If external communications are deemed necessary, then facilities should be put in place to assure the security of these connections. Callback system, appropriate connections devices (modems), alternate communication paths etc. should be built into the system not only to allow secure communication, but also to minimize disruption of services.

Implementation
When the system is implemented, the external communication should be assured to function in a secure and effective manner.

Maintenance
The necessity of external communications should be periodically re-evaluated to determine their continued necessity. Security and the availability of alternate channels should be tested to assure their continued functionality.

Classification: System Security

Method: Restricted Use of System Utility Programs

Description: System utility programs exist on every organizations system. Whether they come with the operating system (such as unix's smit) or they are designed inhouse, their use must be restricted to authorized users.

Several ways exist to restrict permission: the utilities could be placed in restricted directories, their access labels may be configured in such a way that only authorized user can have access to them. If the program was developed by inhouse staff, then a password scheme may be associated directly with the utility, requiring a user to enter a password before being allowed access to it. Such passwords should be regularly changed to prevent password sharing which could lead to misuse of the utility in question.

The advantages of such a system is that it forces programmers and other staff to use accepted means to accomplish their tasks. Unfortunately, this may lower productivity or, worse yet, encourage programmers to develop their own utilities which may be impossible to control. (Parker, 1982)

Logs for all system utility programs should be kept listing their use, the time that they were used, and the name of the user accessing them.

Implementation
Appropriate controls are placed on all utility programs to allow them to be accessed only by authorized personnel.

Maintenance
The entire directory structure is occasionally searched for any "home-made" utility programs. Utility program logs are maintained and scanned to assure that unauthorized access is being prevented.

Classification: System Security

Method: Assign File and Programs to Users

Description: Access labels (see System Security: Data Classification) must be assigned to a file or program with a purpose. In most private industries, access to data on the corporate system is most often associated with job function. (Parker, 1982)

For this reason, user access classification should be attributed based upon job function. The overhead required for determining access rights is decreased in this fashion, and the task of managing permissions is simplified. This also supports the separation of duties methodology (see Organization: Separation of Duties).

The granularity of the operating system or database will determine to what level permissions can be given. For example, if the granularity extends to the field level, then employees in the personnel department may have access to a user name, address and phone number from the employee file, while employees in accounting would be able to read the employees name and salary. Using this method, privacy is strengthened internally, as well as externally.

If the granularity only extends to specific data files, then the responsibility for maintaining and guaranteeing the availability of a file or program can be assigned to a specific user, in effect distributing the responsibilities of managing the system by assigning files and programs to whom they most pertain.

The advantages of this system is that it forces the necessity of collusion in order for fraud to be successful, and this collusion is more difficult, as offenders can be more easily identified. Browsing through files is also prevented, therefore privacy is strengthened.

User authentication (see System Security: User Authentication) must be secure for this plan to work, if an intruder can "spoof" an employee, then the security gained by this method will be severely compromised. (Parker, 1982)

By using groups to determine file and program access, we can bypass one of the major disadvantages of Separation of duties. If an employee is missing, a task can still be completed by someone in his/her group.

Analysis
The employees that will have access to the system are classified into groups by job function.

Design
The operating system structure is designed with the job function groups in mind. Directories can be dedicated to a job function, thus further isolating files and programs from intruders.

Implementation
The system is implemented and tested by carefully monitoring initial system use. Users must be able to accomplish their tasks without undue difficulty but privacy and security must still be maintained.

Maintenance
As new data and programs are assigned to the system, the permissions involved must be carefully considered and monitored. Users should be queried as to their satisfaction with the system. Do they have access to all the data and programs they need? User productivity must be carefully monitored to assure that security is no more counter-productive then it needs to be.

Classification: System Security

Method: Data Classification

Description: On a secure computer system, sensitivity labels must be associated with each subject and object under its control. These labels, sometimes called access labels or classification levels, are used as the basis for any mandatory access control decisions. The OS or DBMS must maintain label integrity, these labels should be maintained even if data or programs are exported to another machine. (DOD, 1983)

Even human readable output should have it's data classification associated with it. File dumps, reports and forms should have their access labels included in the hard copy to avoid data from leaving the organization in hard copy form.

Each user must also be assigned a classification label. It is this label that will determine the user's ability to access data and programs on the system.

Database management systems can be used to manage data classification issues. The granularity of the database refers to the smallest data size that can be directly assigned a classification label. The granularity of the database can, and should be used to enforce Separation of duty limitations (see Organization: Separation of Duties).

In a DBMS, a subject acts on an object. A subject can be any application, query, update, security procedure etc.. while an object is a data entity. It may be a field,

tuple, table or database. Subjects are considered active, while objects are considered passive in nature. A classification level is assigned to an object when it is created, while a subject's classification is inherited from the user who spawned it.

There exists 3 strategies that can be used to implement a secure DBMS (Laferriere, 1990). These are:

(1) the trusted filter (TF).

(2) the balanced assurance (BA) method.

(3) the uniform assurance (UA) method.

The trusted filter is perhaps the easiest to implement. The filter acts as an intermediary between the DBMS and the rest of the system. No data enters or leaves the DBMS without passing through the filter. In this way, the DBMS need not be a trusted system. The trusted operating system in conjunction with the trusted filter can enforce all security issues and maintain all necessary audit logs.

To be effective in its task, it is important for the trusted filter to always be invoked, never be bypassed, be tamper proof (ie. secure) and be small enough to be thoroughly validated (Laferriere, 1990).

To enforce security regulations, the trusted filter attaches access labels to each tuple. The labels are bound to each using a cryptoseal mechanism. The label then becomes a

part of the tuple. Whenever a tuple is extracted from the DBMS, the filter verifies the attached labels to ensure that the data has not been modified or corrupted. If a label does not have the appropriate value, the operation is aborted and an appropriate audit log entry is generated.

The advantages to the trusted filter method are many. It is hardware independent so it may be implemented on a regular DBMS or a Distributed Database Management System (DDBMS). It requires relatively little overhead and can be maintained very easily. The primary disadvantages with the trusted filter approach is its limitations. Because the trusted filter is external to the DBMS, its granularity is not as fine as the other two approaches, being at a tuple level. Its security properties are also restricted to data that passes through it and as such it cannot monitor static data that resides in the database for a long period of time.

The second approach is the balanced assurance method. This method offloads some of the security consideration to the trusted operating system. By using the operating system's labelling abilities, it creates several "data containers" of varying classifications to hold the data. The operating system is also responsible for maintaining audit logs. The advantages to this system is that it requires very low overhead. The disadvantages to the system are the relatively heavy granularity resulting from the operating system's labelling.

One of the main failings of this system is with the use of many multi-level tables. If uni-level tables are used, the containers can be simply labelled, and the data retrieved from the containers quite easily. The widespread use of multi-level tables (a table that contains tuples classified at different security levels) will result in tuples stored in different containers. This will greatly increase the overhead necessary to update and retrieve the data.

The final approach is the uniform assurance method. This method is very popular in organizations today. Simply stated, all security and auditing responsibilities belong to the DBMS. The DBMS, which must be trusted, will be responsible for the labelling of all data items, the evaluation of security levels and the maintenance of audit logs. This system has very fine granularity. Indeed DBMSs often support data labelling at the field level, allowing very sophisticated queries to be developed. The disadvantage of this system is that it requires a good deal of overhead, and the audit logs could not be merged with the operating system's audit logs.

Despite this, there is no question that the uniform assurance approach is the best suited to produce a highly secure system and not simply a collection of secure subsystems. (Laferriere, 1990) However, if security at this degree is not necessary, the balanced assurance model is suggested for uni-level tables and the trusted filter approach for multi-level tables.

Analysis
The data necessities of each job function are ascertained. User productivity is key at this stage. User must have access to the data necessary for them to do their jobs, but no more. Privacy and security are of paramount concern and data should be assigned to user using the "least-priviledge" principle.

Design
The granularity necessary to implement an appropriately secure system is determined. Once determined, this information can be used in conjunction with other information (multi or uni-level? OS or DBMS ?) to determine what scheme will be used to manage access labels.

Implementation
The system is thoroughly tested to assure that access labels are being properly managed and effectively implemented. Data levels must remain intact even if the data is moved from one machine in the network to another.

Maintenance
Data label integrity must be assured with every modification to the system. Access levels to new data must be ascertained and must regularly audited to assure continued appropriateness.

NOTE: The following method will not be included in the final framework. It was included here in order to provide a description of the method and it's purpose.

Classification: System Security

Method: Bell-Lapadula Data Rule (McHugh and Thuraisingham, 1988)

Description: (For a discussion of access labels (or classification levels) see System Security: Data Classification)

When a subject attempts to act on an object, certain rules are applied to determine its success.

Bell and LaPadula suggested two rules to govern the interaction of subjects to objects. The first, called the simple security property, states that if a subject's security classification is higher than or equal to an object's security classification, then it may read from that object.

The second rule, called the *-property, states that a subject may write to an object providing that its security classification is equal to or below that of the object.

As shown in figure 1, the only subject that would be able to both read and write to an object would be one that has the same security classification as the object. Although this may seem inflexible, there are some advantages to this system. First of all, by

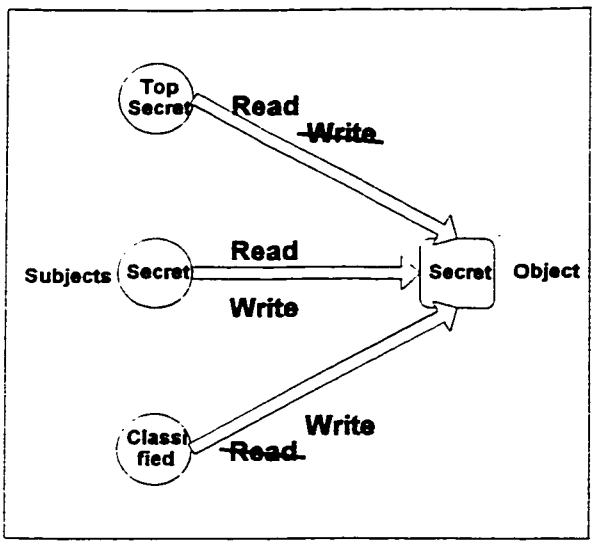


Figure 2. Simple security and *-property

allowing subjects to write to objects that are higher than they are, a system high audit log can be kept. By placing this log at the highest level its security will be assured, as only the system administrator should have access to the highest level.

Despite this fact, the *-property is often only partially implemented. Often, it is modified so that a subject can write only to objects of the same level. Whether the *-property is fully or partially implemented depends on the DBMS being used.

This set of rules has proven very effective at managing data access, though it's use should be limited to organizations that deal with large amounts of data of different sensitivity levels.

Design
The necessity of the implementing the Bell-Lapadula data rule must be ascertained. If this data rule is desired, then a DBMS which will allow the application of this rule should be acquired, as complete and effective inhouse management of this rule is beyond the technical means of most companies.

Classification: System Security

Method: Technical Review of Operating System Changes

Description: Any changes to the operating system or to the basic work environment, whether they be made by the manufacturer or inhouse, should be carefully scrutinized. All details of the change must be reviewed, and their impact on security, current work practices and applications.

Operating system (OS) changes must be assured to compromise neither control or integrity of the system. Inhouse OS changes should be even more closely scrutinized to assure that the changes made will not conflict with vendor updates. (Parker, 1982)

Maintenance
Changes to the operating system or system work environment must be carefully monitored in order to assure that they do not compromise the following areas: security, worker productivity, application performance and system integrity.

Classification: System Security

Method: Cryptographic Protection

Description: Encryption is the transformation of data into a form unreadable by anyone without a secret decryption key. (Fahn, 1993)

Initially, compression utilities often doubled as cryptographic utilities. The algorithms used by these utilities served a dual purpose. Not only did compressing the data increase the efficiency of data storage, but it also converted the data into non human-readable format. (Parker, 1982) The widespread use of these compression utilities has severely degraded their usefulness as encryption tools.

Encryption can be used in many situations (Ruder and Madden, 1978):

- **Remote Encryption:** If an encryption mechanism exists at a central facility, all systems which communicate with the central one, whether by direct network communication or by dial-up lines, must also support the same encryption methods.

- **Encryption for Transport:** The use of encryption when transmitting data to a third party. The third party must have access to facilities in order to reverse the encryption process.

- **Communication Encryption:** Encrypting all communication lines, including network lines, between inhouse systems.

- **Encryption of Data on System:** The use of encryption on all data, or merely sensitive data, on an organizations computer system will help assure information privacy.

- **E-Mail Encryption:** In widespread use today, email often contains sensitive information. The overhead associated with encrypting an email message is low, considering that encryption facilities are built into most available email packages.

There are essentially two main cryptographic schemes available today: public-key and secret-key. (Fahn, 1993)

Public-key cryptography was invented in 1976 by Whitfield Diffie and Martin Hillman. In a public-key system, each person gets a pair of keys: a public key and a private key. The public key is published and used to encrypt any mail or other data sent to the owner of the public key. The owner, upon receipt of the document can then use his/her private key to decrypt the message. (Fahn, 1993)

The major advantage to this system is that the private key need **never** be revealed, thus making it much more secure.

A simple mathematical explanation of the public-key scheme (Vaughn, Saiedan and Unger, 1993):

1) D(E(P)) = P

2) E (D(P)) = P

3) Cannot deduce D from E

4) E Cannot be broken by a plain text attack

Therefore all may give E, retain D.

Where:

E - Encryption Algorithm

D - Decryption Algorithm

P - Plaintext Message

In secret-key encryption both the sender and receiver of the data must know the same secret-key. This key is used both to encrypt and decrypt the data. The impact of this scheme is that the key must be communicated, making it more vulnerable to discovery by an outside source. However, secret-key cryptography tends to be considerably faster than public-key. (Fahn, 1993)

A third option is available which combines the strengths of the first two. In this methodology, the message is encrypted using secret-key cryptography for speed, and the key is encrypted using public-key cryptography before it is communicated to the

second individual. In this way the speed of secret-key is achieved, and the threat of an outside source discovering the key is minimized using public-key.

Examples of public-key encryption tools include RSA and PGP which allow both encryption and authentication of the transmitted data. The main secret-key system is the Data Encryption Standards (DES).

Another advantage to encryption is that it allows the use of the digital signature. The digital signature is often as good as it's more mundane counterpart when used to determine the legitimacy of an electronic document. The use of a proper digital signature guarantees that the document cannot be repudiated by the original sender.

A digital signature consists of two parts. The first is a method of signing a document such that forgery is infeasible, and the second is a scheme to verify that the signature was actually generated by whomever it represents. (Fahn, 1993)

A problem with many networks involve communication from one area of the network to another. Specifically, a network must be able to assure that instructions or data coming from one network have the authority to reside or take action on another. The question is: how does one element of the network authenticate an incoming request from another area of the network? This problem has been solved by Kerberos. Kerberos is a secret-key network authentication scheme that uses DES for encryption and authentication. It

is designed to authenticate requests for network resources rather than to authenticate the authorship of documents. Kerberos provides real-time authentication in a distributed environment but does not provide for future third-party verification of documents.

Kerberos uses a dedicated site on the network, called the Kerberos server, to perform centralized key management as well as administrative functions. The server maintains a database which holds all the keys necessary for authentication on the network, including the secret-keys of all users. It also generates session keys whenever two or more users wish to communicate securely, and authenticates the identity of a user who requests certain network services.

The main problem with Kerberos is that, like other secret-key systems, it requires the use of a trusted third party. In this case, the Kerberos server takes that part. If the server is ever compromised, the integrity of the whole system will fall apart. (Fahn, 1993)

There is already a considerable body of legislation dealing with the use of cryptography. It should be carefully reviewed before any cryptographic tools are considered.

Analysis
The sensitivity levels of the data or determined. If no sensitive data exists on the system, then cryptographic measures need not be implemented.

Design
If it is determined that cryptographic measures are necessary, then the function that these measures will fulfil must be determined. Remote encryption, encryption for transport, communication encryption, encryption of data on system, email encryption or network authentication. If one encryption scheme can be used to fulfil all encryption requirements, then one method should be chosen and used throughout the system. This will minimize the overhead necessary to manage the encryption procedures. If network authentication is necessary due to the presence of sensitive data or procedures, a Kerberos server can be establish to manage the security of network affairs. The choice of other cryptographic tools depends on the needs of the organization. If speed is of the essence, DES can be used. If flexibility and security are of priority concerned, then RSA may be a better choice. A combination of both can also be used.

Implementation
Once encryption tools have been implemented, their use must be carefully monitored. Users should not be able to circumvent the encryption tools, but the tools should not represent an undue strain on their productivity.

Maintenance
New cryptographic techniques should be monitored. There is a great deal of work being accomplished in this area, and new tools are emerging constantly. If a tool is developed that will have a significant impact on the system, then it should be considered for use. Legislative issues should also be monitored to assure that the use of cryptographic tools by the organization does not contradict any laws.

Classification: System Security

Method: User Authentication

Description: A central tool of any computer security system, regardless of it's goal, is the ability to uniquely identify each user of the system. (Chalmers, 1986; Boockholdt, 1989)

If an individual is capable of successfully "spoofing" another individual, then not only will the spoofer able to access information he/she has not been cleared for, but all the audit trails will point to the compromised user.

In most systems today, the main method of user authentication is the user identification/password combination. This is most often used because it most frequently comes packages as part of the operating system. There is therefore no incremental costs involved in implementing such a scheme. However, password can often be difficult to manage. If passwords are administered centrally, the person issuing them could easily compromise them. On the other hand, the individual who is responsible for selecting his/her own passwords often chooses words that are obvious (last name) or easy to guess (password). There are checkers to help minimize the second problem by verifying the password against a standard dictionary and rejecting it if it is too easily guessed. Users can choose effective password simply by combining inserting digits in a word (passw0rd1) or misspelling a word (pazzword), but few users are motivated enough to go to the trouble. (Chalmers, 1986)

Password systems, despite their widespread use, are often very vulnerable. Knowledgeable users often find it very easy to get their hands on the system's password file. In some cases, the passwords are even kept in plaintext (Wong, 1987). If a password system is to be used, the file containing the password should be protected from users. Encryption of the file is the minimum precaution that should be taken. Isolating the file from all but administrative users on the system is offers a higher level of security. (Parker, 1982)

A password should be assigned to each user. Group passwords expose the system to unnecessary risk, particularly if the group has special, or administrative privileges (Boockholdt, 1989; Parker, 1982). In addition (Parker, 1982; Ruder and Madden, 1978):

- password should be changed whenever an employee leaves the company
- supervisor or special passwords should be changed very frequently
- users should be allowed to change their password as they like
- failed logins should be documented for audit. The failed password should **not** be included in the audit log, as the presence of a plaintext file with misspelled passwords will greatly increase the risk to the system
- the operating system should force users to change their passwords at a predetermined time period (eg. every six months).

New technologies are available that will make user authentication much more effective than password systems, and nearly impossible to circumvent. Among these are: fingerprints, signature dynamics, retinal scan, keycards etc. Most of these are major improvements, but their high cost ($500-1000 per terminal) discourages their widespread use. (Chalmers, 1986; Ruder and Madden, 1978)

Passwords that become invalid after their first use, known as one-time passwords, are also an option, but once again more research must be conducted on their use before they become practical and inexpensive enough to be adopted by a large number of commercial systems. (Chalmers, 1986)

Analysis
The level of security needed for the system is determined. The higher the level of security necessary the greater the need for effective user authentication.

Design
The method of user authentication is decided upon. If high levels of technology are to be used (keycards, retinal scan etc.) then a systematic review of all the technologies available should be conducted to find the one most suited for the application based on effectiveness and cost. If a password scheme is selected, facilities should be implemented to allow users to change their own passwords as they wish, but also to verify the password selected to assure that the user has not chosen a password that will be too easy to guess. Audit logs for failed login should be established so that attempts to "hack" into a users account can be identified. Password files should be encrypted and stored in such a way as to prevent user from browsing.

Implementation
Users should be instructed to change their default passwords as soon as the system comes online. A knowledgeable user can take advantage of a default password to "spoof" another user and gain access to facilities that should be restricted to him/her. Technological user authentication schemes should be carefully explained to users, along with any precautions that should be taken to avoid any unnecessary risk (eg. report lost or stolen keycards).

Maintenance
The periodic password changes should be enforced and a password checking mechanism should be occasionally run against all user and system passwords to assure that they are difficult to guess. Technological user authentication methods should undergo regular testing to assure their continued working order.

Classification: System Security

Method: Automatic, Timed Terminal Logoff

Description: Software systems should exists to automatically logoff users after a terminal has been inactive for a specified period of time. (Wong, 1987; Ruder and Madden, 1978) Many network operating systems provide this functionality.

Design
Software should be written or purchased to support this feature if it does not already exists within the shell or operating system.

Implementation
This facility should be activated within the shell or operating system. Programs which were written or purchased must be implemented and tested.

Classification: System Security

Method: Billback system

Description: Each user's daily, weekly, monthly or annual charge is calculated as if he/she was paying for the resources that they consumed. This is not a real charge, and the user is never expected to pay it. By reviewing the charges for system usage, unusual usage can be identified. The charges can be further subdivided into time of day, time of month etc... to determine if the pattern of system usage is consistent with predicted or expected usage.

Projects can have a resource total assigned to them upon completion to help predict resource consumption of similar projects in the future. The billback system allows system administrators to plan ahead for necessary resources, and to determine unusual resource consumption patterns that may result from system abuse or penetration. (Wong, 1987; Ruder and Madden, 1978)

Design
If the operating system does not provide the facilities for a billback system, then software will have to be written inhouse or purchased.

Implementation
The system is tested online. The overhead necessary to manage such a system must not significantly impair system performance.

Maintenance
Periodically, the billback system logs must be carefully scanned to determine regular resource consumption patterns. Any deviation from these patterns should be closely scrutinized to determine the cause.

Classification: System Security

Method: Hardware Monitors

Description: Similar to the billback system (see System Security: Billback System), hardware monitors record levels of channel usage throughout the system. Network connection usage, external connection usage and any other hardware that can be monitored for usage. The log created is then compared with historical and predicted usage to determine any unusual resource consumption.

Design
If the hardware does not provide the facilities for a monitoring itself, then specialized hardware will have to be purchased to perform this function.

Implementation
The system is tested online. The overhead necessary to manage such a system must not significantly impair system performance.

Maintenance
Periodically, the hardware monitor logs must be carefully scanned to determine regular resource consumption patterns. Any deviation from these patterns should be closely scrutinized to determine the cause.

Application Security

Method	Description	Cost
Production Program Authorized Version Validation (Parker, 1982)	Version information in order to differentiate an authorized application from an unauthorized application.	
Responsibility for Application Program Controls (Parker, 1982)	Responsibility for controls in application programs should be assigned to concerned personnel. Documentation of controls is critical.	
Program Quality Assurance (Parker, 1982; Ruder and Madden, 1978)	Applications must work as they are documented, all program modifications should be documented, and an independent organizational body should be responsible for testing programs to assure their proper functioning.	
Secrecy of Data File and Application Name (Parker, 1982)	Users of the application programs do not need to know the actual file name nor the application name.	
Programming Library Access Control (Parker, 1982)	Must restrict access to source code.	
Input Data Validation (Parker, 1982)	All data entered should be validated.	
Processing Time Controls (Ruder and Madden, 1978)	Restricting programs to certain times of execution.	
Well-formed Transactions (Clark and Wilson, 1987)	Based upon accounting principles. The well formed transaction is used, in tandem with Separation of duty, as an effective organizationally based security tool.	

Classification: Application Security

Method: Production Program Authorized Version Validation

Description: The name and other identifiers (version number, modification date, file size etc) of a program being executed is compared with a list of authorized copies in order to determine that the program is authorized to run on the system. This severely restricts the ability of trojan program to be executed unnoticed.

In order for this to succeed, an authorization file must be kept and maintained each time a new program is added to the system or an existing program undergoes some modification. This adds additional complexity to the maintenance and production running procedure. In some cases it may have to be disabled for emergency or recovery procedures. Test facilities will have to be kept on a separate machine or in a separate program library in order to avoid this procedure. (Parker, 1982)

Design
As programs are written, they are catalogued in the permissions file along with their identifiers (modification date, file size etc.). The runtime must be modified to read this and validate any program before it is executed. The validation file should be encrypted and kept in a safe directory to avoid modification. If this file is corrupted or modified in any way, it could not only render any validation useless, it could also lead to denial of service for legitimate jobs.

Maintenance
Any program modifications or the introduction of a new program must lead to a new addition or modification of the file. This process should be automated so that when a program is copied into an active program directory by an authorized individual, the changes are immediately made to the validation file.

Classification: Application Security

Method: Responsibility for Application Program Controls

Description: The inclusion of controls in application programs should be explicitly stated and documented starting with the system analysis phase and continuing through the other 3 stages of SDLC. The responsibility for the effectiveness of controls should therefore be shared among many people.

EDP auditors, who participate in the analysis and design stages, systems analysts, programmers, users and data owners, must all be aware that they are responsible for monitoring and maintaining the effectiveness of application controls. The controls should be thoroughly documented to ensure the proper completion of their implementation, test, development of operational procedures to carry out the intent of the controls as well as to ensure their integrity during change and maintenance. (Parker, 1982)

Each person must be aware of the reasoning and nature of his/her responsibility. Effective application controls depend on each person carrying out their tasks.

Analysis
Any controls that are necessary for proper application security are documented by the systems analyst. The EDP auditors participate in the definition of controls at this stage.

Design
Programmers use the documentation of controls (see analysis) in order to design and build applications that will run securely on the system. The flow of program controls are documented.

Implementation
User are made aware of their rights and responsibilities for application program controls. Proper behavior on the system is explained to them, as is any control that may effect their productivity or otherwise hinder the rapid completion of their task. Data owner are told that regular validation of their data is necessary to ensure that the controls are still effective.

Maintenance
Documentation is periodically reviewed and compared to existing controls to ensure that all is still functioning according to original specifications. If a problem is located, the user whose responsibility is most closely associated with the problem is expected to take responsibility for resolution (e.g. if it is a coding problem the programmer should make the change; a design problem would lead to either the programmer or systems analyst, a behavioral problem would be attributed to the user etc.). Any modifications to the controls in an application program will lead to an appropriate modification to the controls documentation.

Classification: Application Security

Method: Program Quality Assurance

Description: Whether software is produced for resale or for inhouse applications the quality of the software should be verified. This is particularly critical when the software is responsible for the safety of humans or animals.

A testing or quality control (QC) group should be established to independently examine all programs and related documentation produced to ensure their accuracy before production use or resale. This activity is best authorized by software development management or by the QC department. (Parker, 1982; Ruder and Madden, 1978)

Excessively formal program development standards should be avoided. Basic life-cycle procedures should be established and accepted by programmers before more elaborate practices are required. However, although the guidelines to programming should be relatively minor, they should be stringently enforced.

All changes to programs should be logged in a permanent document, which can then be used to assess approval of the changes. (Parker, 1982)

Design
Once the programs are completed, they must be thoroughly tested by an independent group (QC) to ensure their correctness, completeness and robustness before they are implemented on the system. The functionality of each program should be documented to facilitate its testing.

Maintenance
Any changes made to a program, or any programs added to the system, should be thoroughly documented and undergo QC testing to assure the accuracy of the change, or the correctness, completeness and robustness of the new program.

Classification: Application Security

Method: Secrecy of Data File and Application Name

Description: The name of programs and data files need only be known in certain cases. Computer program development, documentation, job setup and, in some cases, computer operation are tasks that would need program and file names. However, users that are in a transaction relationship (i.e. are involved with the computer system strictly through production programs) need not be concerned with the actual file names, and can address their programs by alternate names (ie. menu selection).

This precaution denies users who gain access to the operating system the explicit knowledge necessary to modify individual programs or data files. The least-privilege principle should be used for the dissemination of such information in order to reduce exposure to a sensitive asset. This also facilitates the separation of duties methodology (see Organization: Separation of Duties), which implies separation of information as well. (Parker, 1982)

Analysis
Users are classified into at least two categories based upon their relationship to the system. Those that manage the system, document, write programs versus those who have a transaction relationship to the system.

Design
Precautions should be taken with management to assure that user are given system information (i.e. file and program names) on a least-privilege basis.

Implementation
User are given only the minimum information needed to be able to effectively use the system as their job description mandates.

Maintenance
Users, particularly those who have access to sensitive information, should be periodically reminded of the least privilege rules. Sensitive information should be prevented from easy dissemination. Individuals should be aware of which information is sensitive and which is not.

Classification: Application Security

Method: Programming Library Access Control

Description: Program libraries often contain a wide range of information. Not only is the source code for the current production programs located there, but also source code for programs in development or that are currently being tested.

It would be simple for a knowledgeable intruder to modify the source code of any of these programs, so that when it was compiled and executed it would perform a function it was not originally designed for.

Because of this, the program library should be physically separated from other activities. No user other than legitimate developers, programmers, documentation or quality control personnel should have access to the source code, and their access should be closely monitored. Program accuracy, documentation and controls are particularly important if the programs are being written for resale because of the strict contractual limitations and liabilities. (Parker, 1982)

Classification: Application Security

Method: Input Data Validation

Description: Validating data entered via program for accuracy and legitimacy is important. Not only will it help prevent the GIGO syndrome, but it will also prevent knowledgeable users from taking advantage of the lack of validation by entering data meaningless to the actual transaction, but meaningful to the operating system, file structure etc.

Validation should include examination for out of range values of data, invalid characters in data fields, exceeding upper and lower limits of data volume, and unauthorized or inconsistent control data. In certain cases the actual meaning of the data can be checked against other data entries in order to determine if the entry is consistent with past entries of a similar nature.

Although validation creates a fairly high degree of overhead, early error detection will prevent error propagation, help the entire system run more effectively and efficiently. (Parker, 1982)

Design
As programs are being designed, appropriate validation sequences should also be considered for each data entry the program requires. Validation types (range checks, validating an entry against a set of valid entries etc.) should be considered, making sure to allow users to enter all necessary data. The validation should aid users in entering valid data. It should not hinder user by preventing them from determining what data is valid. Meaningful error messages are very effective way of helping user enter valid data.

Implementation
The validation techniques used should be monitored to assure that they are correct and complete. They should, along with meaningful error messages, help users enter accurate data.

Maintenance
Validation techniques may have to change if terminology or ranges change. Transaction procedures should be closely monitored to determine any changes that may affect the way that validation works.

Classification: Application Security

Method: Processing Time Controls

Description: Application can be given authorized times of execution, limiting their use to ceratin times of the day, or days of the week etc. This can prevent unauthorized use of production programs during off-hours.

The duration of batch program run can also be restricted. If tests show that a batch program takes an average of 10 minutes of execution, then controls can be implemented to prevent it from taking longer than the established time for execution. This way the program is of limited use as a Trojan, having a restricted runtime. (Ruder and Madden, 1978)

There should exist a method of disabling this control for emergency or recovery purposes.

Design
Programs are classified according to the time they are permitted execution. Times of day, days of week, month, or even year can be specified. The runtime can then read this file, which can double for a version validation file (see: Application Security: Production Program Authorized Version Validation), and determine whether or not the program is authorized to run at this time. The actual execution time can be restricted in a similar way.

Maintenance
New programs must be added to the file.

Classification: Application Security

Method: Well-formed Transactions

Description: Users should not be permitted to manipulate data arbitrarily , but rather in constrained ways that serve to ensure the integrity of the data. Much like double entry bookkeeping, well-formed transactions are a series of transactions that complete a particular task. In double entry bookkeeping, issuing a check includes an entry to the cash account as well as a matching entry in accounts payable. If one entry is not performed, then the system does not balance. This can be detected by an independent test. (Clark and Wilson, 1987)

Well-formed transactions operate much the same way. The software that issues checks would not be permitted to issue a check and modify the cash account unless an entry were already made to the appropriate accounts payable account. If both of these well-formed transactions were performed by the same person, then fraud would be possible, but by combining the concept of well-formed transaction with separation of duty (see Organization: Separation of Duty) fraud becomes very difficult.

If each well-formed transaction is performed by a different individual, then collusion would be necessary to defraud the company. In our check-issuing example, the person responsible for accounts payable and the person responsible for issuing checks would have to cooperate for fraud to work. Of course, a task composed of 30 separate

transactions could require as many as 30 people to collude in order for fraud to be possible.

A very common mechanism to further prevent fraud, is to record all data modification so that actions can be audited later. (Clark and Wilson, 1987)

Analysis
Business procedures are closely analyzed and broken down into logical transactions. Rules are then developed to validate these transactions against each other (ie. will not perform transaction B unless transaction A has been made).

Design
Software is then written to support these rules, and to perform the well-formed transactions on the data files. Users should never be allowed to modify the data files directly. The software will have to be assigned to job functions keeping separation of duty in mind. A user is not permitted to perform two sequential well-formed transactions in the same transaction chain.

Implementation
Users are instructed as to the logic behind the well-formed transaction rules, in order for them to properly understand their role in the transaction chain. The rules are carefully scrutinized to ensure that they do not severely hamper user performance.

Maintenance
Any modifications to job functions or to transactions will lead to a modification of the rules, and therefore to the software. Rules modification should be closely scrutinized to determine the impact that the modification will play on other well-formed transaction rules in the system.

Standards

Method	Description	Cost
Compliance with Laws and Regulations (Parker, 1982)	The system must fall within legally imposed guidelines.	
Participation of User at Critical Development Times (Parker, 1982)	User information should be used to design controls.	
Program Standards (Ruder and Madden, 1978)	Policies and procedures to ensure that all programs follow accepted programming standards.	

Classification: Standards

Method: Compliance with Laws and Regulations

Description: A document should be prepared for any new or existing system. All relevant laws and regulations dealing with computer security, information privacy and any industry specific legislation that may be of concern, should be outlined. The systems compliance with these laws should also be stated. To assure the legal validity of the document, legal council should be consulted. (Parker, 1982)

Analysis
All laws that would affect the system under design will be catalogued.

Design
The system is designed with the legislation in mind. A document is produced outlining all pertinent laws, as well as the system's compliance to them.

Maintenance
New legislation should be monitored to assure that the status of the system does not change. Modifications may have to be made to the system to assure that it complies with these new laws. The compliance document should be modified as new laws emerge, even if the system already complied to these laws.

Classification: Standards

Method: Participation of User at Critical Development Times

Description: Computer users, including those providing data or using computer output, should be involved in determining the standards for controls in the organization.

Explicit control requirements can be specified by users to systems analysts and programmers. These are then scrutinized by auditors to determine that the controls stated are complete and correct (see Audit Role: Application System Design Verification). Users statements on controls can then be used to define general control standards throughout all applications on the system.

Users are in the unique position of actually using the applications on the system. They are often aware of what controls are needed to not only make the environment more secure, but to make their jobs easier (e.g. validation). In this way, the accountability and responsibility for controls can be shared between analysts, designers, programmers and users, helping to ensure their completeness and correctness.

Users, being affected by every control implemented, should agree with the necessity of controls. Involving them in controls specifications is a strong step in this direction. (Parker, 1982)

Analysis
Users are queried as to the strengths and weaknesses of the present system's controls. Suggestions for improvements are solicited.

Design
Controls are designed, keeping in mind the comments and suggestions of users. The controls are then explained to the users to verify that they meet with their approval. If users are unhappy with the controls, the necessity of these controls should be reviewed with them. If the control is necessary, it will remain intact. If, however, changes can be made to accommodate the user's interest, they should be.

Maintenance
Users are periodically queried on the completeness and effectiveness of controls. Their responses are reviewed to determine if additions, changes or deletions to the system's controls should be made in order to make the system more secure, or easier to work with.

Classification: Standards

Method: Program Standards (Ruder and Madden, 1978)

Description: Maintenance and debugging are greatly facilitated if the program was written using accepted industry or organizational standards.

These standards are often initially adopted by organizations from industry standards, and then evolve over time. They include controls, input, output, screen layout, report layout, processing, file i/o and other standards. Standards are determined early and remain relatively stable. A document, listing all accepted standards, is then produced and distributed to all programmers.

Software exists, or can be written to verify that programmers are following the accepted standards.

Analysis
The programming standards to be followed are compiled and/or reviewed to determine their applicability. Software is purchased or written to verify that source code follows the established standards.

Design
Once the standards have been detailed, the software is written. Software cannot be placed in the central software library unless it follows the organization's standards.

Maintenance
If standards change (this should happen infrequently), the standards document is modified appropriately, highlighting the changes, and distributed to the programmers affected.

Audit Role

Method	Description	Cost
Audit Logs (Clark and Wilson, 1987; Parker, 1982, Ruder and Madden, 1978)	Audit logs should be kept for every important activity of the system. The cost in overhead should be balanced with the increase in security gained.	
Independent Computer Use by Auditors (Parker, 1982; Ruder and Madden, 1978)	Isolation of auditing system from system being audited will lead to less downtime and more processing available for the audit function.	
Requirement and Specification Participation by Auditors (Parker, 1982)	EDP auditors should be involved in the design of important application systems to ensure that controls are adequately specified.	

Classification: Audit Role

Method: Audit Logs

Description: The auditors main function is to gain assurance that all actions within the organization fall within established risk levels. This means that they must identify and take appropriate action on any behavior which would go against established security controls, policies or procedures.

Audit logs are an effective way of identifying such behavior. Though they require some overhead, and are not foolproof, audit logs can be kept for a great deal of system occurrences. Ideally, an audit log would record every action taken by a user on the system. The overhead associated with such a log prohibits this, however. Some suggested logs include, but are not limited to (Ruder and Madden, 1978):

- Operator (superuser) Log which records all actions taken by an operator on a system, or any user which is grated operator privileges.
- Sensitive file modification logs
- System crash log which will help in identifying the reason for a system crash
- Application program change log
- Improper logon log
- Channel volume log (see System Security: Hardware Monitors)
- External network command logs (see System Security: Firewalls)

These logs should be audited frequently to determine any unusual or suspicious behavior on the system, regardless of the source. Ceratin precautions should be taken. Logs should be kept for a minimum period of time, so that they can be referred to as historical information. Printed logs should have page numbers associated with them to prevent tampering. All logs should be date and time stamped. (Parker, 1982)

Automated verification procedures should be established whenever possible. Logs should be automatically, as well as manually, audited. Any exception reports generated by the automated procedures should be reviewed frequently, such as on a daily basis. Whereas the logs themselves can be audited on a weekly or monthly basis, depending on their volume. Elements that can be identified by automated procedures include activities outside normal working hours and access to file not normally associated with the job function. The greater the degree of granularity, the greater use the logs will be to the audit function. (Chalmers, 1986)

Analysis
Any actions that should be logged are identified.

Design
The procedures to implement logging on the system are designed with the system, and written to work in tandem with any other policies and procedures used to secure the system. The overhead created by such logs must not affect system performance significantly. Automated log scanning procedures should be designed to scan for all possible and pertinent activities, and produce an exception report available to the auditors and the computer security officer.

Implementation
Log procedures and files must be tamper proof. Their use is severely limited if users can easily modify them.

Maintenance
Logs and exception reports must be reviewed frequently. Automated procedures should be modified if any pertinent actions can be automatically scanned. Auditors must also assure that the logs and logging procedures are not tampered with.

Classification: Audit Role

Method: Independent Computer Use and Audit Tools by Auditors

Description: A separate computer system should be used to perform major system and/or application audits. This serves many purposes: it allows auditors to become familiar with procedures involved in installing applications, and using the system, it reduces prevents undue overhead on the main system, it allows verification of the portability of applications and prevents tampering with audit procedures. (Parker, 1982)

Tools should always be kept isolated from production systems to prevent tampering. Internal auditors should be the only personnel with access to these tools. (Ruder and Madden, 1978)

Although the cost of this is may be prohibitive to smaller organizations, the increase in audit security will justify the cost in larger organizations.(Ruder and Madden, 1978) If a separate system cannot be implemented, audit procedures should be isolated from all other system activities.

Analysis
If an audit system is not already operating. It should be designed to parallel the new system.

Design
Although on a smaller scale, the directory structures should be similar to ease the transfer of files, application and audit logs. The operating system and control procedures should be identical on both systems.

Implementation
A system audit should be conducted shortly after the production system is implemented to assure its accuracy and completeness.

Maintenance
Any change to the production system should also be made to the audit system to assure accuracy of the audit procedures.

Classification: Audit Role

Method: Requirement and Specifications Participation by Auditors (see Standards: Participation of Users at Critical Development Times)

Description: Auditors should be involved in the development of any new system and critical applications. In this way they can assure that proper audit procedures are supported and that system and application controls are adequate. Auditors should be required to sign any formalized application or system specifications.

This assures that controls to be implemented on systems fall within organizational specifications.

Analysis
Auditors are involved in the analysis of the new system in order to identify any control weaknesses.

Design
Auditors must verify any proposed controls to determine if they are sufficient to reduce risk to acceptable levels. This applies both to application and system programs. The final design should contain the signature of the auditor associated with the development of the system as a confirmation of approval.

Chapter 3

Validation of the Framework

The Questionnaire

The Commercially Viable Computer Security Implementation Framework that resulted from the extensive literature review is both comprehensive and vast Each of Fine's 9 categories was closely evaluated to determine which methods would be most appropriate at addressing each category's security concerns. Encompassing 53 security methods, the framework attempts to address all security issues which may affect a commercial organization.

This framework, once validated, could prove to be of great worth to the information technology field, stating explicitly what steps are necessary to implement security methods at each phase of the SDLC. This allows the information system professional to design a system in parallel with the security measures that will eventually protect it from any threat deemed significant. As previously stated, security methods implemented at the outset are far more effective at deterring computer abuse then security methods implemented as an afterthought (Vaughn, Saiedan & Unger, 1993).

Due to its large size, validation of all elements was beyond the scope of this paper. In order to reduce the scope of the project, we confined our validation to one of the nine pillars: system security. This pillar includes but is not limited to: a network firewall to

protect organizational data from external threats; remote terminal physical security; restriction of systems utility programs; file and/or program assignments; data classification; the Bell-Lapadula data rule; technical reviews of operating system changes; cryptographic protection; user authentication; automatic, timed, terminal logoff; hardware monitors and bill back systems. Of the nine categories or pillars, this one was chosen because it has received the most industry and media attention, and should therefore be familiar to most of our respondents.

The questionnaire (see Appendix A) includes one page of questions for each method under scrutiny as well questions about the overall security of the system under consideration and organizational parameters of interest.

The questionnaire's instructions includes a description of the system development life cycle to ensure the respondent's familiarity with the terminology used throughout the questionnaire. The respondent was asked to keep a particular computer system in mind when answering the questionnaire questions. This computer system which, when considering our sample base, was doubtless an inhouse machine developed by the corporate information systems department, had to fit the following criteria:

- it must be networked (inhouse or connected to an external network such as the Internet)
- it must support more than one user
- it may be composed of any combination of hardware or software

The following page shows the questionnaire questions for the Firewall security method.

Firewalls

Dedicated hardware and software to prevent unauthorized actions, used exclusively in network environments.

A- Please rate your familiarity with firewalls by circling the appropriate number, where 1 corresponds to very unfamiliar and 5 corresponds to very familiar.

Very Unfamiliar 1 2 3 4 5 Very Familiar

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of firewalls as if applied to your system. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive

C- Please indicate to what extent you agree or disagree with the following procedures for developing firewalls at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates complete agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

- **Analysis:** When the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered..... ____

When the system is deemed very sensitive to the organization, then the network connection is not attempted..... ____

When a firewall is required, any operations that need to be available between the system (or network) and the untrusted network are determined, and the firewall is designed to allow these operations..... ____

- **Design:** When a network connection is deemed necessary, then the firewall is designed prior to the system..... ____

- **Implementation:** Once the system under design is connected to the firewall, the firewall is evaluated to assure that it provides appropriate security to the system..... ____

If the system remains vulnerable, then it should be disconnected from the untrusted network immediately ____

- **Maintenance:** To ensure that required security levels between the untrusted network and the organizations system are being maintained, the firewall must be audited regularly..... ____

D- When developing a firewall for your organization, to what extent were the above tasks followed? Circle N/A if a firewall was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A

The first questionnaire question, question A, seeks to establish the respondent's familiarity with firewalls. The standard scale used throughout the questionnaire is a 5 point ordinal scale. Used with a variety of anchors, this scale allows a suitable granularity to the responses. In the case of question A, the scale has "Very Unfamiliar" and "Very Familiar" as the anchors, allowing the following possible answers: very unfamiliar (1), somewhat unfamiliar (2), marginal familiarity (3), somewhat familiar (4) and very familiar(5). The instructions on the questionnaire clearly state that if the respondent indicates a familiarity level of 2 or less, the remainder of the questions regarding this method should be skipped. This assures us of meaningful responses to the remainder of the questions on the page.

Question B was an evaluation of the perceived effectiveness and cost of firewalls. Each attribute was evaluated using another 5 point ordinal scale. In order to evaluate effectiveness, the anchors used are "Very Ineffective" through "Very Effective", with the cost element using "Very Inexpensive" through "Very Expensive", thus maintaining the standard "negative to positive" flow of the scale used throughout the questionnaire. These two questions are particularly important to commercial organizations. The resulting cost/effectiveness ratios made possible by these questions (for ratios see the System Security section of the framework) can be used to justify security expenses to management.

If anything, the E911 document valuation (Sterling, 1992) illustrates the inability of management to give a proper value to data. In the same way an objective valuation of security measures is very complex. One can measure the cost of a firewall by the expense of the hardware, software, cabling and consultant fees that was necessary to implement it. However, costs can also include lost productivity suffered by employees wrongly blocked by the firewall, lost billing from clients whose access or email was stopped by the firewall due to domain conflicts, and any other unplanned problems that may occur when implementing such a large-scale network filter. Finally, the cost of support staff necessary to maintain the firewall over the years is also difficult to valuate at the onset.

With all these elements, conducting an objective valuation proved an insurmountable obstacle. Indeed, it is a fitting subject for future research. As such, we settled for a subjective valuation by the respondent. It is only once such security methods have been implemented and maintained for a few years that the "true cost" of a security method can really be ascertained.

Question C was not a single question. Rather, it had a variable amount of statements which corresponded with each task that must be accomplished during each phase of the SDLC when implementing firewalls (see the framework for a description of these tasks). The respondent was asked to indicate his/her agreement with each statement.

The statements were clearly subdivided according to each phase of the SDLC: analysis, design, implementation and maintenance.

Once again a 5 point ordinal scale is used with the anchors being "Completely Disagree" and "Completely Agree" with the respondent writing his/her selection in a blank following each question.

The series question in part C contains perhaps the most significant questions in the questionnaire, being more directly related to the validation of the framework than any other. When looking at the framework, we see that the statements made for each method on how implementation should be approached at each stage of the SDLC were transformed into a series of basic, single premise statements for question C. For example, in the framework section on Firewalls, we have the following implementation suggestion at the analysis stage of the SDLC:

"If the system under development will be connected to an untrusted network,especially the internet, then the use of a firewall will be considered. Any operations that need to be available between the system (or network) and the untrusted network is determined, and a firewall is designed to allow these operations.

The sensitivity of the system is evaluated, and if the system is deemed very sensitive to the organization, then the network connection is not attempted."

This statement was transformed into three individual statements in part C of Firewalls:

- **Analysis:**

When the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered..... ___

When the system is deemed very sensitive to the organization, then the network connection is not attempted..... ___

When a firewall is required, any operations that need to be available between the system (or network) and the untrusted network are determined, and the firewall is designed to allow these operations..... ___

In this fashion, the respondent is asked very specifically, to evaluate every statement made in the framework, and state whether or not they agreed with this implementation schedule.

Finally question D, the last questionnaire question specifically on firewalls, is used to determine whether or not the respondent had implemented firewalls and, if so, whether the respondent had followed the implementation schedule outlined in question C. A 5 point ordinal scale with "Not Followed at All" and "Followed Completely" as anchors was used to evaluate to what degree the implementation schedule was followed. The scale was accompanied by a second option, N/A, which was used to indicate that a firewall was not implemented on the system under consideration. The responses to this question are used to determine whether the implementation schedule outlined in question C is actually used in organizations, as well as to determine what methods are usually implemented in a commercial environment.

In addition to security method specific questions, the questionnaire includes several other questions. The first of these questions, question 101 (see Appendix A: Questionnaire and Demographic Information of the Sample) ascertained the overall perceived effectiveness of their security measures. Once again the scale was a 5 point ordinal, with the anchors being "Very Ineffective" and "Very Effective".

The questionnaire ended with a series of questions whose purpose was to determine certain key attributes of the organizations. These include: number of employees, number of users on the system in question, the respondent's involvement in the development of the system, whether the system was purchased or developed inhouse, what scheme was used to develop it and, finally, what the main user base is for the system in question.

The Sample

Previous research in Concordia University's Decision Science and Management Information Systems department had resulted in a mailing list of 485 managers of corporate information systems from across Canada. Using this list as the frame of our sample, a questionnaire (see Appendix A) was sent to only one manager per company. Two weeks after the questionnaire was mailed, a follow-up letter requesting a prompt response was sent in order to maximize the return rate.

Of the 485 questionnaires sent, 31 completed questionnaires were received, giving us a response rate of 6.4%. We believe that the low response rate was due to the relatively long length of the questionnaire, which contained 107 questions, as well as the sensitivity of the issues addressed. Despite the promise of anonymity and the assurance that only compiled data would be published, the information system field is still wary of reporting any evidence of system security problems (Fuentes, 1997).

Findings

As stated, the majority of the questionnaire was composed of 5 point ordinal scales designed to determine to what level the respondent agreed or disagreed with a particular statement:

Question C: Completely Disagree 1 2 3 4 5 Completely Agree

or slight variations on the form:

Question B: Very Ineffective 1 2 3 4 5 Very Effective

Very Inexpensive 1 2 3 4 5 Very Expensive

Due to the low response rate, each of these 5 point ordinal scales was broken into a 2 point binary scale, with answers of 1 or 2 transformed into a response of 1, and responses of 3 or greater becoming 2. Responses of 3 were included with the upper end of the scale to reflect the lack of a negative answer, in much the same way that a familiarity of 3 was considered sufficient to respond further to the question under consideration.

Question C: Disagree 1 2 Agree

Question B: Ineffective 1 2 Effective

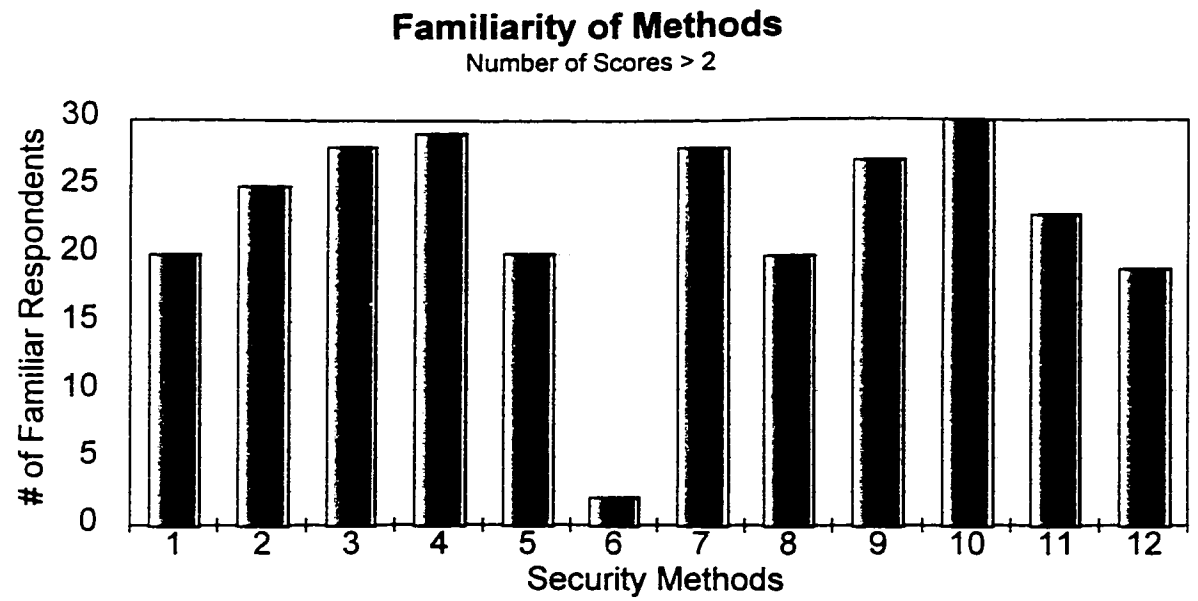
Inexpensive 1 2 Expensive

When comparing two variables of interest, the resulting 2X2 table was used to generate a chi-square result. A chi-square result table for lower sample sizes was specifically generated to determine the p-value for resulting chi-square values (for chi-square generation program, see Appendix B).

In many cases the results of the chi-square tests were not significant. The reason for the lack of significance was a strong weight of samples in one or two of the quadrants. These cases will be clearly indicated in the analysis of the findings.

Analysis of the Findings

Questionnaire question A: Familiarity of Methods



1-Firewalls	2- Remote Terminal Physical Security	3-Restricted Use of System Utility Pgms
4-Assign Pgms and Files to Users	5-Data Classification	6-Bell-Lapadula Data Rule
7-Tech. Review of OS Changes	8-Cryptographic Protection	9-User Authentication
10-Auto. Timed Terminal Logoff	11-Bill Back System	12-Hardware Monitors

Figure 3. Familiarity of Methods

In the initial analysis of the questionnaire results, a few things became immediately apparent. If we look at familiarity as a whole, we see that an average of 22.5 respondents were familiar with any given security method, and yet the Bell-Lapadula showed a familiarity of only 2. The general ignorance of this security method can easily be explained by its relative obscurity and need for technical expertise in its application. It is not surprising that a method that dictates the reading and writing of data objects (a

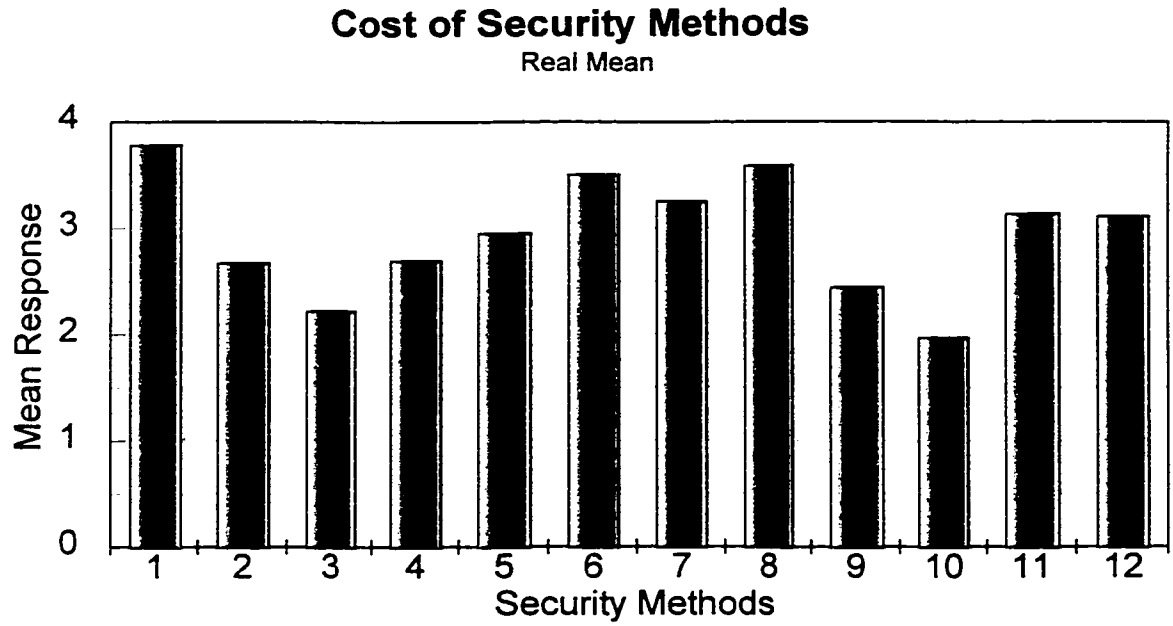
task usually relegated to a database management system) is unfamiliar to most respondents. Few organizations have need of such granularity of data object control, preferring to leave the management of such tasks to third party database management systems.

The top five security methods, based on the familiarity of respondents with these methods are (the number in parentheses being the percentage of respondents familiar with the method): automatic, timed terminal logoff (100%); assigning file and programs to users (96.7%); restricted use of system utility programs (93.3%), technical review of operating system changes (93.3%) and user authentication (90%).

This high level of familiarity comes as no real surprise. Automatic, timed terminal logoff being widespread today with the use of password-enabled screen savers. The assignation of files and programs to users as well as user authentication are obvious on most network systems, demanding a login prompt and subdividing directories based on username. Finally, restricting system utility programs and scrutinizing operating system changes is common sense. Ignoring changes to the operating system platform is a sure recipe for disaster, particularly if the organization uses inhouse software, and allowing untrained end-users access to system utility programs is akin to handing a child a loaded gun.

Despite the high levels of familiarity with these methods, we will see that they are not all equal in the eyes of IS managers. Some methods are used more than others, and some are perceived as more expensive or less effective.

Questionnaire question B: Cost/Effectiveness



1-Firewalls	2- Remote Terminal Physical Security	3-Restricted Use of System Utility Pgms
4-Assign Pgms and Files to Users	5-Data Classification	6-Bell-Lapadula Data Rule
7-Tech. Review of OS Changes	8-Cryptographic Protection	9-User Authentication
10-Auto. Timed Terminal Logoff	11-Bill Back System	12-Hardware Monitors

Figure 4. Cost of Security Methods

The perceived cost of the various security methods ranges from 1.97 (Automatic, Timed Terminal Logoff) to 3.78 (Firewalls) with an overall mean of 2.94. This tells us a few things. The first is that security measures are considered, on the average, costly to the organization. These costs can range from a direct dollar cost to more intangible

costs such as lost productivity and increased system overhead. It is for this reason that we insisted on a subjective valuation of costs.

The complexity involved in implementing and maintaining a firewall justifies its relatively high standing in terms of cost. The scope of this method is much greater than any of the others, with the exception of Cryptographic Protection and the Bell-Lapadula Data Rule which came in second and third with scores of 3.59 and 3.5 respectively, though it should be noted that the results of latter are based on responses from only two professionals. Firewalls regulate virtually all electronic traffic entering or leaving the organization's internal systems. It is only natural that its perceived costs are higher.

On the other side of the spectrum, the methods with a narrower scope, such as automatic, timed terminal logoff; restricted use of system utility programs and user authentication, ranked significantly lower with scores of 1.97, 2.214 and 2.44 respectively. These three methods can typically be enabled from within the operating system, either by setting appropriate permissions on certain files or by implementing passwords and inactivity timers. In reality, the final implementation of these methods may be as simple as checking a box on a system administration window.

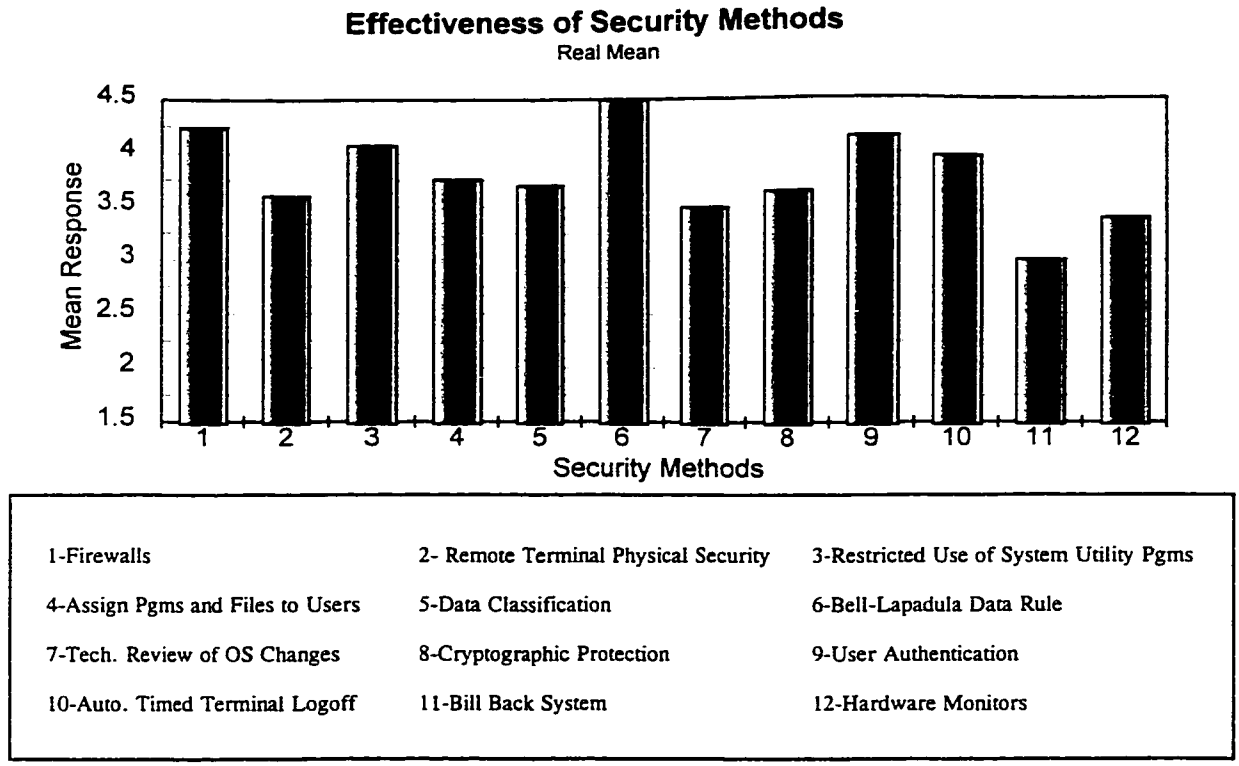


Figure 5. Effectiveness of Security Methods

The results of the questionnaire reveal that the most effective security method is the Bell-Lapadula Data Rule. While this statement is reasonable considering that the Bell-Lapadula Data Rule determines the read/write properties of any process on the system and therefore protects the validity of data by eliminating any non-authorized data modification, the result as listed above is not necessarily reliable when we remember that only two respondents were familiar with the method to begin with. However, it is safe to say that the method is well respected among those familiar with it.

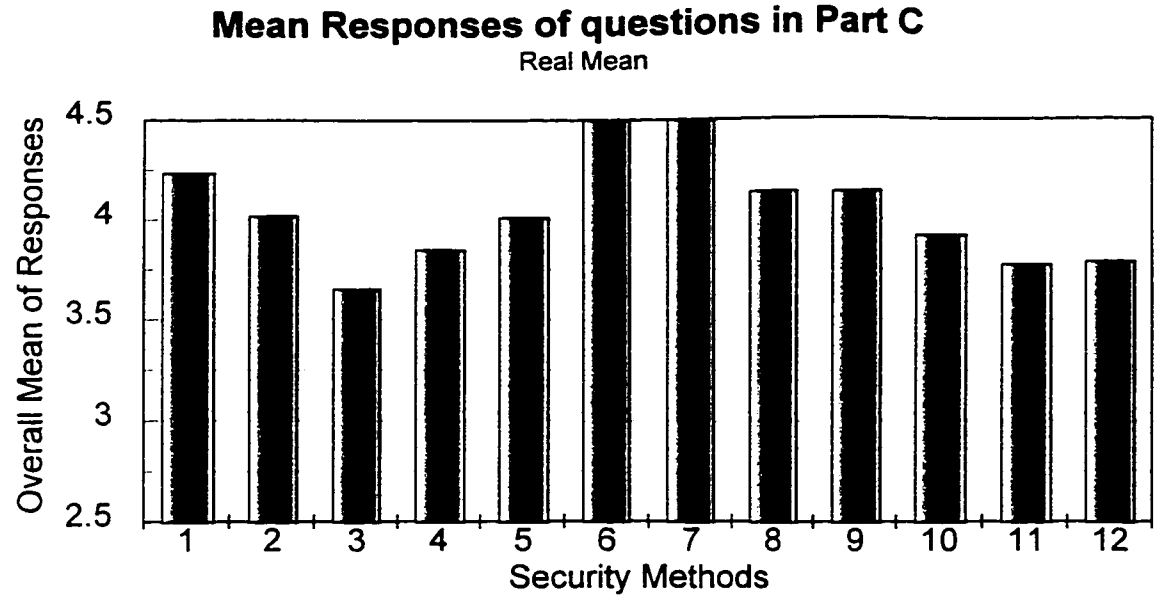
A close second is the Firewall method. Once again, no surprise when we consider the prevalence of the Internet in today's computing environment. Any organization

wishing to take advantage of the information or communication possibilities of the earth's largest computer network must be prepared to defend themselves against intrusion. Those not willing to invest in a firewall should steer clear of the Internet or prepare themselves for data loss, denial of service and numerous other pranks, hacks, attacks or cracks.

User authentication, restricted use of system utility programs and automatic timed terminal logoff also scored quite high with 4.15, 4.07 and 3.97 respectively. These three methods are commonly used in today's organization and there is a reason for this. Together, they form the basic security structure for a system. The first, user authentication, presents a "door" to the user in the form of a login prompt. If the user has the "key" or password, then that user may access the system. The second, restricted use of system utility programs, keeps the user honest by only allowing access to system features that they would normally need. Finally, automatic, timed terminal logoff, automatically shuts the system "door" behind the user, preventing others from "walking" into the system through an opened door. Each person on the system must have their own login or "door", their own password or "key" and, of course, the system must make sure to close the "door" if the user forgets.

No method listed was considered ineffective. With a mean score of 3, Bill Back System was the lowest scoring of the twelve methods, but can still be considered effective.

Questionnaire Question C: Validating the Implementation Schedule



1-Firewalls	2- Remote Terminal Physical Security	3-Restricted Use of System Utility Pgms
4-Assign Pgms and Files to Users	5-Data Classification	6-Bell-Lapadula Data Rule
7-Tech. Review of OS Changes	8-Cryptographic Protection	9-User Authentication
10-Auto. Timed Terminal Logoff	11-Bill Back System	12-Hardware Monitors

Figure 6. Mean Responses to the questions in part C

Part C contained the most direct questions regarding the framework validation. The results reveal that the framework itself seems valid in terms of steps that must be completed in order to implement a given security method on a system at the onset of system development. The overall mean was calculated by determining the mean response to each question, and from this determining the overall mean for part C. The lowest mean response occurred for Restricted Use of System Utility Programs, which

scored a 3.65, indicating that overall respondents were in agreement as to the implementation schedule.

The strongest responses occurred in the Bell-Lapadula Data Rule and Technical Review of Operating System Changes, both scoring a 4.5 as mean response. Again it should be noted that the mean for the Bell-Lapadula Data Rule was calculated from only two responses. However, with an overall mean for all the methods under consideration of 4.04, it is safe to say that according to industry professionals, the suggested implementation schedules were valid.

Questionnaire Question D: Past Use of the Implementation Schedule

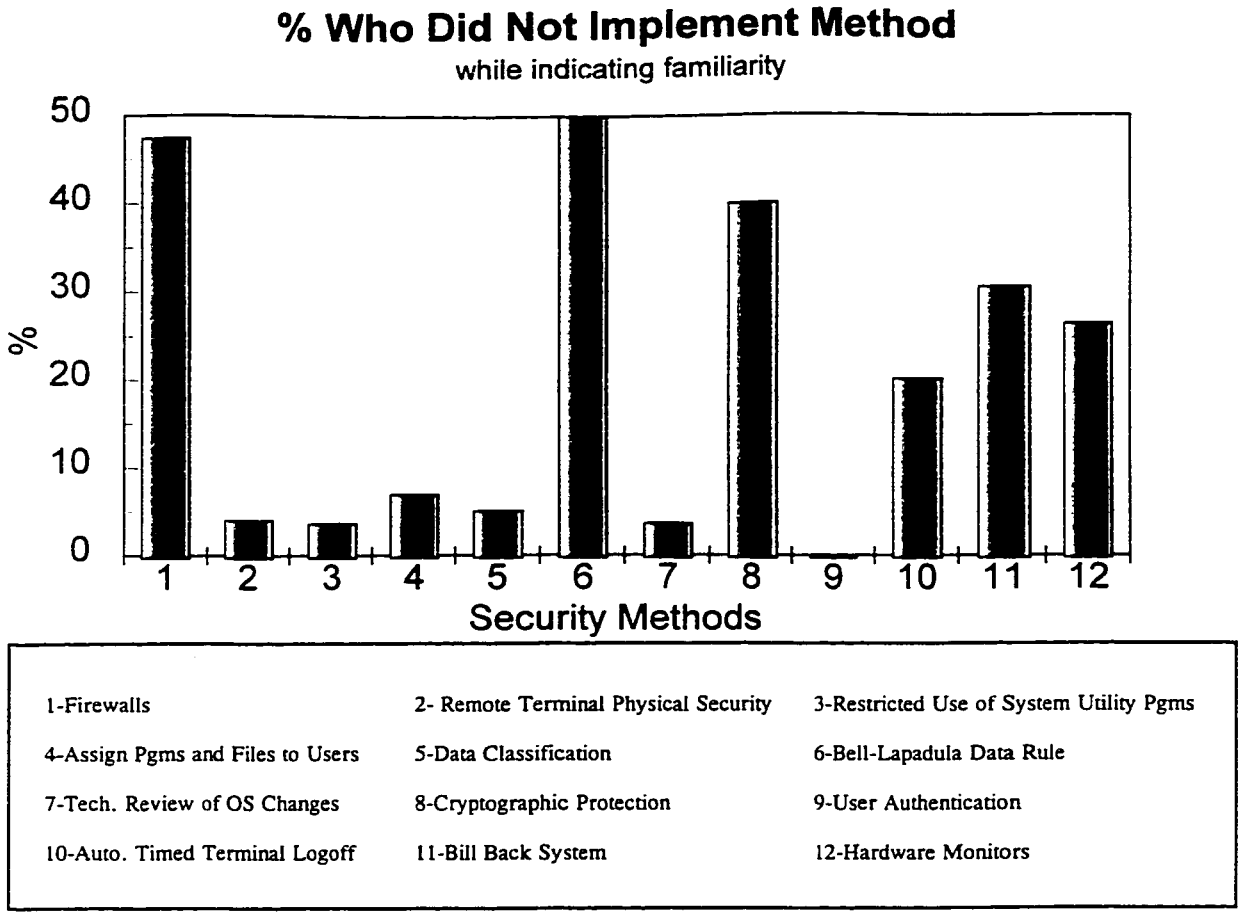


Figure 7. Percentage of respondents who did not implement the security method while indicating a familiarity with it.

Question D had a peculiarity associated with it that was absent from the previous questions. If the respondent was familiar with the security method, but had not implemented it on the computer system under consideration, a "not applicable" (N/A) answer was given. By analyzing the number of N/A answers, we see that in most situations, if a respondent was familiar with a method, then that method was implemented in the system. Indeed, you may notice that the User Authentication

method is missing from the above chart, the reason for this is that 100% of respondents that indicated a familiarity with the User Authentication method had applied it to their system.

Most of the methods are extensively used by those familiar with them. Fully 50% of the 12 listed methods have implementation rates of 93% and above. Certain methods that scored high, such as firewalls, the Bell-Lapadula Data Rule and Cryptographic Protection are only useful in cases where the system is connected to an untrusted network or the need for data security and a high level of granularity is present, and as such are not expected to be used in all circumstances. It is therefore reasonable that they have relatively high levels of non-implementation.

Other methods, such as Automatic Timed Terminal Logoff, Bill Back Systems and Hardware Monitors scored 20%, 30% and 26% respectively. These relatively high scores reveal that there are certain areas which could be quite effective at deterring computer abuse, but are largely ignored by the industry. Reasons for this could be that those in charge of computer security at the site may think that the other methods cover a sufficient amount of security leaks and that the risk level of the system has already been reduced to an acceptable amount. This is a completely justified decision.

For some methods such as Bill Back Systems and Hardware Monitors, the method may seem outdated or "lo-tek" and are thus deemed ineffective. A decision to ignore a

method because it is no longer the "latest and greatest" is an unjustified decision. In the war against computer crime, security personnel must use any tool at their disposal to ensure the security of the system. Bill Back Logs and Hardware Monitors measure the basest units on a computer system: cpu usage, disk usage, memory usage and hardware resource usage. Once a baseline of "usual" consumption levels has been established, identifying idiosyncracies can be one of the most effective methods of identifying unauthorized system use.

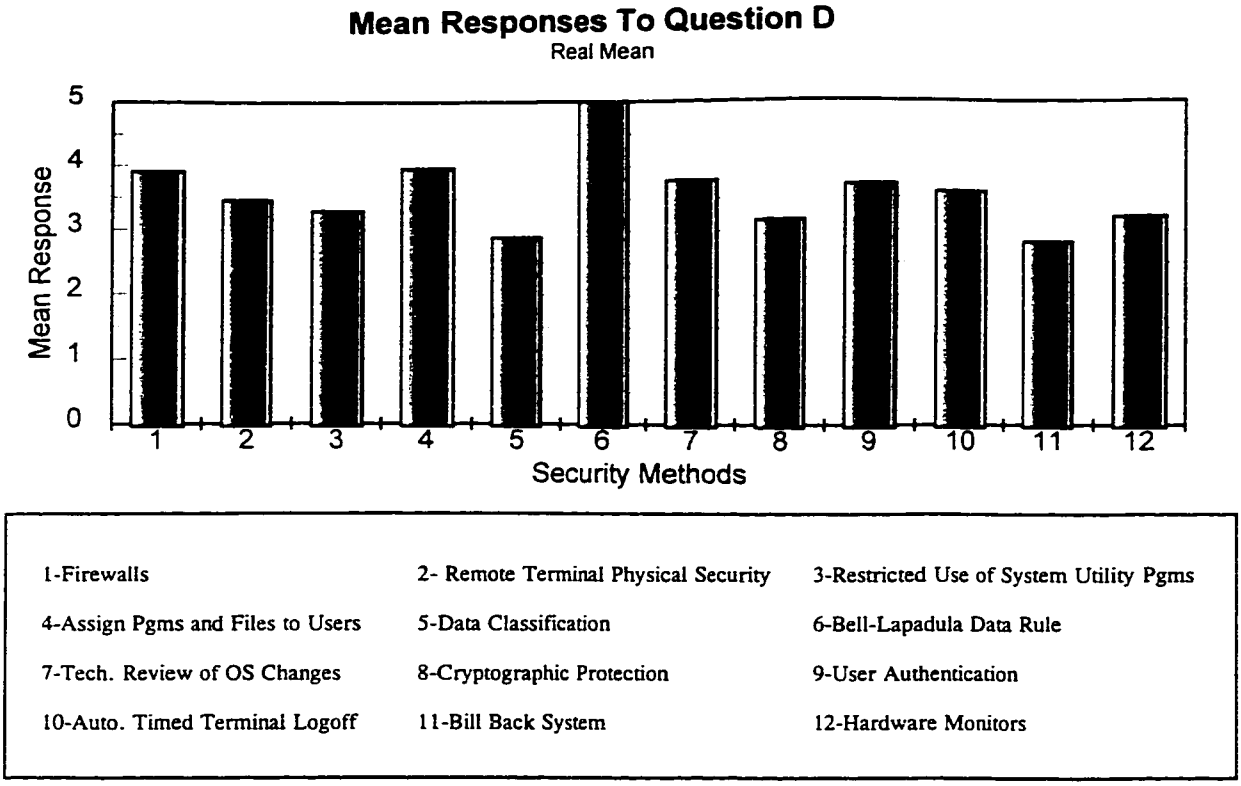


Figure 8. Mean Responses to Question D

Of those that were familiar with the method in question, and did apply them on their computer system, a question remains: did they follow the implementation schedule listed in Question C?

Figure 7 indicates that, for a great majority of respondents, the answer is "yes". The overall mean of question D, when a response was given, is 3.53. This high overall mean tells us that in the majority of situations, the implementation schedule was followed when implementing the method under consideration. The highest result was for the Bell-Lapadula Data Rule, with a mean level of 5 (based on the two responses

received), indicating in all occurrences of implementation of this rule, the schedule suggested in C was followed exactly.

With all other responses falling between 2.81 (Bill Back System) and 3.9 (Firewalls), we can conclude that, in general, the implementation schedule was followed quite closely for each of the 12 methods under discussion.

Specific Questions

Now that we have reviewed the findings of each question individually, we now look to the relationships between questions.

The first three questions are meant to determine the validity of the model as a security implementation tool. The results of the tests are presented in a tabular form. Each cell of the table contains several elements. The first element, in parentheses, is the questionnaire question identification number (QIN). This was necessary because question C contains several different responses, and so a chi-square value had to be calculated for each response. The questionnaire, along with the QIN for each question, appears in Appendix A. Below the QIN the chi-square value is listed when possible. In many cases, a chi-square value is not possible due to the unevenness in the distribution of the responses, which is indicated by n/a. In the parentheses following any actual chi-square value is the p-value of the test, or (ns) if the resulting chi-square value is not significant. Finally, the last entry of each cell is the 4^{th} quadrant weight of the 2X2 matrix.

For example, if we look at the first question, **Q1: If the respondents agreed with the statements made in question C, did they follow it when implementing the security method on their own system?** We compare the results of question C with the results of question D. Since there is only one response to question D and many distinct responses to question C, a chi-square value was calculated for each response to individual

statements in question C vs. the one response to question D, hence the need for question C identification numbers.

Although these question were answered on a 5 point ordinal scale, we transformed this scale into a 2 point binomial scale so that Question C had two possible responses:

Question C: Disagree 1 2 Agree

and question D had 2 possible responses:

Question D: Not Followed 1 2 Followed

The resulting 2X2 matrix looks something like this:

Question C/D	Not Followed	Followed
Disagree	Quadrant 1	Quadrant 2
Agree	Quadrant 3	Quadrant 4

Significance of quadrants:

Quadrant 1: Respondent disagreed with the implementation schedule outlined in Question C and did not follow it when implementing the security method on his/her system.

Quadrant 2: Respondent disagreed with the implementation schedule outlined in Question C but nevertheless followed it when implementing the security method on his/her system.

Quadrant 3: Respondent agreed with the implementation schedule outlined in Question C but did not follow it when implementing the security method on his/her system.

Quadrant 4: Respondent agreed with the implementation schedule outlined in Question C and followed it when implementing the security method on his/her system.

In many circumstances, a chi-square value was not possible due to an abnormal weight in the 4th quadrant. Since there was no relationship in evidence by the chi-square test, but the responses were still favorable because of their location in quadrant 4, the percentage of responses that fell in quadrant four was indicated. As a further example, let us look at the first cell of Table 1:

(4) **n/a** **87%**

This cell describes the comparison between the responses to question D of Firewalls (QIN 11) and the responses of the first statement made in question C of Firewalls (QIN 4) From Appendix A we find that question 4, or QIN 4 is:

Analysis: When the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered..... (4)

We then see that calculating a chi-square value for QIN 4 vs. question D (QIN 11) was not possible (N/A) and, finally that 87% of responses fell in quadrant 4 of the 2X2 matrix, which is probably the reason that chi-square calculation was impossible.

Does the failure to calculate a chi-square value mean that no relationship between the two questions was present? No. If we analyze the abnormal weight in quadrant 4, we can see that 87% of respondents agreed to the following statement: "When the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered" and asked themselves exactly this question when determining whether or not to use a firewall on their system. This kind of mass agreement to a statement was prevalent in the findings, and helps assure the validity of such statements when implementing security methods.

Q1: If the respondents agreed with the implementation schedule from question C, did they follow it when implementing the security method on their own system?

To answer this question, we compared the results of question C with question D. Some transformation of the variables was necessary. In the first place we transformed the 5 point scale to a two point scale in order to generate a meaningful chi-square score. Further, we discarded from this test any questionnaire in which the respondent indicated that they were not involved in the development of the system (question 104).

This analysis will help us not only identify which statements have been used to implement the security method under consideration, but also what percentage of respondents agreed with the statements. As such, both quadrant 4 (agreed with the statement and applied it when implementing the method under consideration) and quadrant 3 (agreed with the statement, but did not apply the method in the suggested manner) will be used to determine whether or not the statement will remain in the final security framework.

Table 1: If the respondents agreed with the implementation schedule, did they follow it? (Question C vs. Question D)

Firewall	Remote Terminal Physical Security	Restricted use of System Utility Programs	Assign Files and Programs to Users	Data Classification	Bell - Lapadula Data Rule
(4) n/a 87%	(15) 1.25(ns) 75%	(24) n/a 82%	(31) .1(ns) 86%	(41) 1.77(ns) 62%	(52) n/a 100%
(5) n/a 87%	(16) 3.95(.05) 85%	(25) .084(ns) 48%	(32) .1(ns) 86%	(42) n/a 62%	(53) n/a 100%
(6) n/a 87%	(17) 9.47(.05) 85%	(26) .745(ns) 59%	(33) .1(ns) 86%	(43) 3.81(.05) 62%	
(7) n/a 87%	(18) .117 (ns) 85%		(34) .290(ns) 73%	(44) n/a 62%	
(8) n/a 87%	(19) 3.95(.05) 85%		(35) n/a 95%	(45) n/a 62%	

Table 1: If the respondents agreed with the implementation schedule, did they follow it? (Question C vs. Question D)					
Firewall	Remote Terminal Physical Security	Restricted use of System Utility Programs	Assign Files and Programs to Users	Data Classification	Bell - Lapadula Data Rule
(9) n/a 87%			(36) n/a 65%	(46) n/a 62%	
(10) n/a 87%				(47) 1.875(ns) 62%	

Table 1: If the respondents agreed with the implementation schedule, did they follow it? (Question C vs. Question D)					
Tech. Review of OS Changes	Cryptographic Protection	User Authen-tication	Auto. Timed Terminal Logoff	Bill Back System	Hardware Monitors
(58) n/a 86%	(63) .563(ns) 55%	(73) .726(ns) 69%	(84) 4.82(.05) 65%	(90) 3.34(.1) 46%	(97) 1.27(ns) 50%
	(64) n/a 66%	(74) .347(ns) 77%	(85) n/a 80%	(91) .929(ns) 46%	(98) n/a 70%
	(65) .563(ns) 55%	(75) .518(ns) 73%		(92) 1.04(ns) 38%	(99) 2.59(.2) 70%
	(66) n/a 66%	(76) .157(ns) 82%			

Table 1: If the respondents agreed with the implementation schedule, did they follow it? (Question C vs. Question D)					
Tech. Review of OS Changes	Cryptographic Protection	User Authen-tication	Auto. Timed Terminal Logoff	Bill Back System	Hardware Monitors
	(67) n/a 66%	(77) .518(ns) 73%			
	(68) n/a 66%	(78) .329(ns) 78%			
		(79) .329(ns) 78%			

Note: Cell format is:

(question number)
Pearson chi-square value.(significance level/p-value (.05, .1, .2, ns - Not significant))
unusual quadrant weight (% of respondents in quadrant 4 (they agree with the statement and implemented the security procedure as suggested)

From these tables we can determine that although many of the chi-square results are missing or insignificant, a mean of 73.15% of responses fall within quadrant 4 of the 2X2 matrix, indicating that a significant majority of the respondents not only agree with the statements made in question C, but that they followed these procedures when implementing the security method in their systems.

The weakest results occur with the Bell-Lapadula data rule. Despite 100% of responses falling in quadrant 4, the number of respondents familiar with the method was only 2. In addition, the method itself is highly technical, and should be purchased rather than

constructed. Due to these shortcomings, the method will be removed from the final framework. The implementation of rules governing the access and updating of data objects on the system should be governed by a trusted DBMS, purchased for specifically this purpose. These results support the statements made by David Clark and David Wilson that, while the military frameworks seek to regulate the control and distribution of classified information (DoD, 1983; CSSC, 1993; Calmers, 1986), commercial sites are more likely to be interested in data integrity (Clark and Wilson, 1987).

Certain areas show weaker results than others. Areas in which fewer than 60% of the respondents fall in the 4th quadrant and no relationships become apparent between the variables of interest according to the chi-square test include: restricted use of systems utility programs, cryptographic protection, bill back system and hardware monitors.

Only 59% of respondents indicated agreement with the two statements made about restricted use of system utility programs. The first statement, question 25, which states that the entire directory structure should occasionally be scanned in order to find any "home-made" utility programs, showed that only 48% of respondents both agreed with this statement and followed it regularly on their system. While this may not be a serious security risk on administrative systems, other systems used as software or hardware development platforms may find that users denied access to certain system utilities can write their own, and that these utilities present a security hole that system

administrators must be aware of in order to properly manage system security(Parker, 1982). In addition, search facilities provided on almost all major operating systems make locating these utilities an easy task that consumes a minimal amount of system resources. For example, the UNIX (DEC) command:

find / -perm -4700 -user root -print

will generate a list of all files that have the setuid bit and execute permission set and that belong to root. This may be the only command necessary to find utilities which could be a threat to the entire system. It takes approximately 10 minutes to complete, depending on the size and load of the system, and can be set up to run automatically every week (day, month...) and generate a log reporting its results.

The next question (26), deals with the periodic review of system utility logs to ensure that the utilities are not being misused. In studies, 50% of computer abuses were discovered through system controls, 41% were accidently discovered, while only 16% were discovered through active detection (Straub and Nance, 1990). Despite this, only 59% of respondents indicated that periodic review of system utility logs was being conducted. Obviously, more attention needs to be paid to active detection of system abuse, such as periodic log scanning. The cost of scanning these logs is minimal, as an automated process could easily be generated to warn system administration of exceptions that occur.

The next method, cryptographic protection, has resulted in two questions returning results of less than 60% of responses in quadrant 4. These questions, 63 and 65, deal with evaluation and necessity of using an encryption scheme. Though only a small number of responses fell in quadrant four, a further 34% or more fell in quadrant 3, indicating that although only 55% applied the suggestions, at least 89% agreed with them.

The results for the first two questions about bill back systems is similar. Though only 46% of responses fell in quadrant 4, when quadrant 3 is added we see that the percentages rise to 76.9% for question 90 and 92.3% for question 91. This indicates that though fewer people applied the suggestions, most agreed with them. Question 92, dealing with periodic log scans in order to identify unusual system consumption patterns, is necessary to make a bill back system effective in deterring computer abuse. Therefore, despite the relatively low results of 69.2% who agreed with the statement, it will remain as it is necessary to make a bill back system effective in detecting computer abuse.

Finally, only 60% of respondents agreed with the first statement pertaining to hardware monitors which states that if facilities to monitor hardware use are not present, they should be purchased. This method will also be preserved as the responses to the rest of the statements in this method were favorable, with 70% of responses falling in

quadrant 4, and studies have revealed that perpetrators who abuse hardware privileges are the most difficult to identify (Straub and Nance, 1990).

Though the majority of users agreed and followed the statements made in question C, this does not mean that the resulting systems were secure. The following test compares the results of question C with question 101, the overall system evaluation. By answering the following question we can ascertain whether those respondents who agreed with the statements made in question C are successfully addressing their own security concerns.

Q2: If the respondents agreed with the statements made in question C, is their system secure?

Objectively measuring computer security effectiveness is a very difficult task (von Solms, van de Haar, von Solms and Caelli, 1994). One reason for this is that computer security includes a great many elements. Do we measure number of security measures implemented? Success rate of these measures? The results or number of a security breaches? The cost of data or services lost or stolen? Is a document that has been taken from your site really considered "stolen" if you still have a copy of the data? (Sterling, 1992)

A comprehensive and objective security effectiveness evaluation is beyond the scope of this paper. Since much of the evaluation of the effectiveness of security is relative, limited by the perceptions of users and managers, we have chosen to use a relative as opposed to objective security valuation. This was accomplished with the following question:

(101)
Please rate the overall effectiveness of the security measures currently implemented on your computer system.

Very Ineffective 1 2 3 4 5 Very Effective

In order to perform chi-square test with the minimal response rate, we further transformed this by dividing it into 2 categories. Responses of 2 or less were classified as unsecure systems, 3 or more were classified as secure systems. Comparing this to the results from question C also similarly transformed, results in a 2X2 matrix. A chi-square result from this matrix is still not always possible due to heavier weights in one or two quadrants. The following table summarizes the results and includes the percentage of responses that fell into quadrant 4 (Agreed with the statement made in question C and has a secure system).

Table 2: If the respondents agreed with the statements made in question C, is their system secure? (Question C vs. QIN 101)					
Firewall	Remote Terminal Physical Security	Restricted use of System Utility Programs	Assign Files and Programs to Users	Data Classification	Bell - Lapadula Data Rule
(4) .053(ns) 90%	(15) .649(ns) 72%	(24) n/a 86%	(31) .513(ns) 70%	(41) .055(ns) 90%	(52) n/a 100%
(5) .263(ns) 75%	(16) .283(ns) 80%	(25) 2.71(.2) 58%	(32) .513(ns) 70%	(42) n/a 95%	(53) n/a 100%
(6) .053 90%	(17) .136(ns) 80%	(26) 4.93(.05) 58%	(33) 1.153(ns) 80%	(43) .117(ns) 85%	
(7) .175(ns) 80%	(18) .283(ns) 80%		(34) .0721(ns) 70%	(44) n/a 95%	
(8) .053 90%	(19) .283(ns) 80%		(35) 6.72(.05) 80%	(45) n/a 90%	
(9) .111 85%			(36) 1.41(ns) 63%	(46) n/a 95%	
(10) n/a 90%				(47) .198(ns) 75%	

Table 2: If the respondents agreed with the statements made in question C, is their system secure? (Question C vs. QIN 101)					
Tech. Review of OS Changes	Cryptographic Protection	User Authen-tication	Auto. Timed Terminal Logoff	Bill Back System	Hardware Monitors
(58) n/a 86%	(63) .368(ns) 76%	(73) .777(ns) 71%	(84) 1.102(ns) 67%	(90) .773(ns) 65%	(97) .091(ns) 63%
	(64) .175(ns) 76%	(74) .376(ns) 71%	(85) n/a 87%	(91) 1.0 86%	(98) n/a 84%
	(65) .175(ns) 76%	(75) 7.53(.05) 82%		(92) .461 73%	(99) 1.97(.2) 78%
	(66) n/a 76%	(76) .173(ns) 82%			
	(67) .175(ns) 76%	(77) 7.53(.05) 82%			
	(68) .368(ns) 76%	(78) 2.24(.15) 82%			
		(79) 2.24(.15) 82%			

Note: Cell format is:

(question number)
Pearson chi-square value.(significance level/p-value (.05, .1, .2, ns - Not significant))
unusual quadrant weight (% of respondents in quadrant 4 (they agree with the statement and have a secure system)

Once again few chi-square results show a relationship between the responses of question C and question 101 due to the weight of responses falling in one quadrant.

The smallest percentage of responses that fell in quadrant 4 was 58% (once again, questions 25 and 26) with a mean of 79.78%. Therefore, despite the lack of chi-square statistics, we can conclude that the vast majority of respondents who agreed with my statements had, in their opinion, secure systems.

There is a step missing however. We have shown that users who agree with the statements made in question C have implemented a great majority of these methods when implementing their systems. We have also stated that users that agreed with the statements from question C have secure systems. It then follows that users with secure system must have implemented their security methods according to the tasks from question C. In order to confirm this, we now compare the responses from question D (the extent to which the tasks from question C were followed when implementing the methods on the user's system) against the result of question 101 (how secure is their system?). This will answer the following question:

Q3: If the respondents followed the implementation schedule suggested in question C, is their system secure?

Table 3: If the respondents followed the implementation schedule suggested in question C, is their system secure? (Question D vs. QIN 101)					
Firewall	Remote Terminal Physical Security	Restricted use of System Utility Programs	Assign Files and Programs to Users	Data Classification	Bell - Lapadula Data Rule
(11) n/a 100%	(20) 1.47(ns) 80%	(27) 5.46(.05) 79%	(37) 3.475(.2) 87%	(48) .616(ns) 58%	(54) n/a 100%

Table 3: If the respondents followed the implementation schedule suggested in question C, is their system secure? (Question D vs. QIN 101)					
Tech. Review of OS Changes	Cryptographic Protection	User Authen-tication	Auto. Timed Terminal Logoff	Bill Back System	Hardware Monitors
(59) .996(ns) 79%	(69) 4.8(.2) 67%	(80) .438(ns) 75%	(86) .414(ns) 76%	(93) .073(ns) 53%	(100) .294(ns) 71%

Note: Cell format is:

(question number)
Pearson chi-square value.(significance level/p-value(.05, .1, .8, ns - Not significant))
unusual quadrant weight (% of respondents in quadrant 4 (they implemented the method according to the statements in C and have a secure system)

Though many of the resulting chi-square values are missing or insignificant in most cases, a mean of 77% of respondents indicated that they followed the tasks described in question C and have built systems which they consider secure. The lowest results occur with bill back systems and data classification with scores of 53% and 58% respectively.

Reviewing the responses given for data classification and bill back systems reveals that in addition to their relatively poor showing in terms of respondents who have secure systems and followed the tasks suggested, they had relatively high percentages of respondents in quadrant 2 (those who have secure systems and did not follow the tasks as proposed in question C). With 37% and 35% of responses to questions 48 and 93 respectively falling in quadrant 2, it seems that few commercial companies enact security techniques to monitor hardware use, or regulate access to data.

As stated earlier, government security models stress a strict and rigid classification of data whereas commercial organizations put less emphasis on the distribution of data and more on the assurance of data integrity (Clark and Wilson, 1987). This explains the disparity between data classification (58%) and assigning file and programs to users (87%), as the latter is meant to assure data integrity, while the former seeks to regulate the secure flow of data. Still the overall responses to the tasks suggested when implementing data classification show a range from 3.75 to 4.37. This leaves us with the conclusion that MIS managers view data classification as an important security measure, but often fail to implement it.

The other method which received relatively low support, the bill back system, is one method of being able to ascertain unusual consumption of system resources. Despite the relatively low score of 53%, the three tasks associated with bill back systems have received a high level of agreement with the respondents. Questions 90, 91 and 92

achieved a mean response of 3.26, 3.70 and 4.39 respectively. Again, this indicates that while respondents agree with the usefulness of the method, few have implemented it.

Cost/Effectiveness

Cost/effectiveness is critically important to commercial security environments. Since commercial environments, unlike military environments, cannot afford to implement every security method available to them, it is important that a proper evaluation of each method include an overall cost versus effectiveness. The following table describes the cost/effectiveness of each of the methods of system security.

The following table lists the respective cost/effectiveness of each method under review. This is number is the mean of each respondents evaluation of costs, divided by each respondents evaluation of effectiveness:

$$\frac{}{\textbf{(Cost/effectiveness)}}$$

The resulting statistic ranges from .2 (1/5) to 5 (5/1), with a statistic closer to zero indicating a higher effectiveness to cost factor.

Table 4:Cost/Effectiveness

Firewall	Remote Terminal Physical Security	Restricted use of System Utility Programs	Assign Files and Programs to Users	Data Classification	Bell - Lapadula Data Rule
0.9444	0.8494	0.5638	0.7883	0.7783	0.8

Table 4:Cost/Effectiveness					
Tech. Review of OS Changes	Cryptographic Protection	User Authen-tication	Auto. Timed Terminal Logoff	Bill Back System	Hardware Monitors
1.0678	1.1111	0.5941	0.5091	1.2384	0.9593

Though most methods seem worth the cost of their implementation, three stand out as providing the greatest return on investment in terms of protection for resources spent. These are restricted use of system utility programs, user authentication and automatic, timed terminal logoff with a ratio of 0.5638, 0.5941 and 0.5091 respectively. This indicates that the perceived effectiveness of these measures is nearly twice their perceived costs.

Facilities for implementing these methods are usually included within the operating system. User authentication and automatic, timed terminal logoff are commonly used in all environments (commercial, military and academic). Implementing a scheme to restrict use of system utility programs has already been discussed in terms of operating system facilities. Their ease of use and implementation, coupled with their relative effectiveness at deterring computer abuse make these 3 choices very attractive. The widespread use of the first two supports these results.

Three more methods stand out with a cost/effectiveness ratio greater than one. These are: technical review of operating system changes, cryptographic protection and bill back system, with ratios of 1.0678, 1.1111 and 1.2384 respectively.

These three methods reflect a greater perceived cost than their perceived effectiveness. Technical review of operating system changes requires a high level of knowledge of the operating system to be able to evaluate what effect the changes will have on software, whether developed inhouse or purchased. Don Parker, the "grandfather of computer security", stated this in *Computer Crime: Computer Security Techniques* (Parker, 1982):

> "Operating system (OS) changes must be assured to compromise neither control or integrity of the system. Inhouse OS changes should be even more closely scrutinized to assure that the changes made will not conflict with vendor updates."

The operating system is the bridge between the hardware and software on a system. As such, it is uniquely vulnerable to attack due to the critical role it plays, as well as the ease with which any potential offender can gain intimate knowledge of your OS. Though in the past, some projects required the inhouse development of the operating system, this is rarely the case anymore. Indeed, only military organizations would have access to the tremendous resources necessary to undertake such a task. Modern operating systems are highly complex and finely tuned pieces of code. Commercially available operating systems range from those specializing in standalone machines, such as Microsoft's Windows 95 and Apple's Mac OS 8, to more complex operating systems

meant for networking several computers together such as Microsoft's Windows NT and the various flavors of UNIX.

With the operating system playing such a critical role in any computer system, close scrutiny of any changes is mandatory. If security and dependability is of any concern to the organization, any new revision or patch must be closely scrutinized to determine the effect it may have on any of the following areas: security, worker productivity, application performance and system integrity. If appropriate steps are not taken, the organization may find its computer services denied to them, or mission critical applications failing, damaging the company's reputation and resulting in lost time, productivity, goodwill and money.

Cryptographic protection is another method which in a high cost/effectiveness ratio. While it is true that the cost of proper cryptographic security is quite high, the results can justify this cost. It is important to note, however, that cryptographic protection is not useful in all situations. An organization must evaluate their need for cryptographic protection, and the area or areas which must be so protected. There are many different areas that cryptography could play an important role: remote encryption, encryption for transport, communication encryption, encryption of data on system, email encryption or network authentication.

If the organization does not have the need for secure communication or transport. The cryptography should definitely be ignored. The cost in resources consumed, time spent, regular review of new cryptographic methods and, in some cases, dedicated hardware make this method expensive and high in maintenance.

Recent questionnaires have indicated that:

> "Over 50% also consider U.S.-owned corporate competitors a likely source [of computer attack]. Over 50% of respondents also cited that information sought in recent attacks would be of use to U.S.-owned corporate competitors. And reflecting the increased competition in the global marketplace, 26% cited foreign competitors as a likely source of attack and 22% also cited foreign governments as a likely source of attack." (Rapalus, 1997)

With information being the new currency among corporations and governments, and with competition fierce in both domestic and foreign markets, many companies view the safeguarding of corporate information as a high priority. In cases such as this, the use of cryptographic protection may not be an option. Appropriate encryption schemes must be evaluated and adopted, as the dissemination of such information could be far more costly to the organization then the cost associated with cryptography.

Finally, the usefulness of bill back systems has already been ascertained. Internal controls are needed to evaluate the possibility of both internal and external threats. Measuring resource use can be an effective way of identifying areas of potential abuse (Wong, 1987; Ruder and Madden, 1978). Since many operating systems already have facilities for implementing a bill back system, and software is readily available for those that don't, the majority of the cost involved in implementing this method is due to the time spent in regular perusal and analysis of the resulting logs. Most respondents

agree that this method can be effective as a tool for identifying abuse, whether the effectiveness of this method justifies the expense involved remains up to the individual organization.

Additional Findings

Q5: Are larger systems perceived as more secure?

To evaluate whether or not larger systems are perceived as more secure, we compare the results of question 103 which evaluates the number of users on a particular system, against the results of question 101, the overall security evaluation.

The results of this test give us a Pearson Chi-Square value of .43750. This value is not significant, and indicates a lack of correlation between the number of users on a particular system and the security of that system. However, 75% of respondents fell into quadrant 4. While we cannot conclude that larger systems are more secure, we see that a vast majority (86%) of systems used in a commercial environment are large, with 50 or more users, and that the same proportion (86%) perceive their system as secure. This belies the results of a 1997 questionnaire conducted by the Computer Security Institute that indicated that 49% of respondents have suffered security intrusions in 1997. This discrepancy may be explained by results from the same questionnaire, which indicated that only 17% of respondents who have been the victim of a computer attack reported the crime (Rapalus, 1997). Although Concordia University is an accredited institution, respondents may not have the same degree of trust in it as they do in the Computer Security Institute, leading to the conclusion that a certain level of non-response bias is responsible for the discrepancy.

Q6: Is there any method of implementation which results in a more secure system?

To answer this question we will compare the responses of question 101 against each response to question 106. Possible answers to question 106 are:

1- Systems Development Life Cycle (SDLC)
2- Prototyping
3- Computer Aided Systems Engineering (CASE)
4- Any combination of the above
5- None of the above

Table 5: QIN 101 vs. QIN 106	
Method of Implementation	**Mean result of 101**
1- Systems Development Life Cycle (SDLC)	3.333
4- Any combination of the above	3.75
5- None of the above	3.5714

We can see from the mean results of question 101, that the implementation method seems to have no significant repercussions to the final security of the inhouse-built systems.

Q7: Are systems designed inhouse perceived as more secure than systems that are purchased?

To answer this question we will compare the results of question 101, the overall security evaluation, with question 105, how the system was developed.

Table 6: QIN 101 vs. QIN 105		
	Non-secure	**Secure**
Inhouse	3	17
Purchased	1	6

The result of the Pearson chi-square test was not significant with a result .9638. We cannot conclude that there is any relationship between the purchase or inhouse development of a system, and the security of that system.

Q8: Do people rate systems they have worked on as more secure?

This will help identify any personal bias based on the respondent's involvement in the development of the system. In order to answer this, we compare the results from question 101, the overall security evaluation, against question 104, which determines whether the respondent was personally involved in the development of the system.

Table 7: QIN 101 vs. QIN 104		
	Non-secure	**Secure**
Was involved	3	23
Was not involved	1	1

The resulting Pearson chi-square value of .13417 fails to prove any relation between the involvement of the respondent and the security of the system. The majority of the respondents were involved in the development of their system, however, and it is likely that the results may be different with a larger sample size.

Chapter 4

Conclusion

The suggested framework for implementing system security methods is sound, with a vast majority supporting the statements made in question C of the questionnaire, which represents the suggested implementation schedule of each measure.

Possible exceptions to this include: cryptographic protection, bill back systems and hardware monitors, restricted use of system utility programs and data classification due to slightly lower results in one or many tests.

Cryptographic protection was one of the least implemented methods and showed a relatively high cost/effectiveness ratio compared to other methods. It's use should be carefully considered for any organization. There is little doubt that it's effectiveness is useful to many commercial organizations, with information becoming more valuable (Rapalus, 1997). Unfortunately the overhead necessary to manage such a scheme may make it prohibitive to some commercial environments. It is important to properly evaluate information security needs in the organization in order to successfully identify an organization whose data is sensitive enough to merit a widespread cryptographic scheme. The results for this measure coincide with previous research that indicated that commercial organizations are more concerned with data integrity than data privacy (Clark and Wilson, 1987). This method will remain in the framework, as it is effective

in it's task, though it's applicability will have to be ascertained for each individual organization.

Though hardware monitors are rarely used in the industry, most respondents agreed that this method is useful and effective in detecting computer abuse. This method will remain in the framework.

Restricted use of system utility programs is rarely used in the industry and is not considered important in determining the overall effectiveness of security of the system, and yet is considered the second most cost/effective security measure in this pillar of the framework. In a recent questionnaire conducted by the Computer Security Institute, 47% of respondents reported attacks from *inside* the organization (Rapalus, 1997). It then follows that restricted access to important system utility program, such as facilities to edit logs or user profiles, change password, shutdown the system, or modify network privileges make a good deal of sense (Parker, 1982). This method will remain in the framework.

Data classification is rarely implemented in commercial organizations today. The main focus of most of the military models (Clark and Wilson, 1987), this method is of marginal use to many commercial organizations today. In spite of this, it remains a relatively cost/effective method of maintaining data security on a site, as most operating systems provide some basic facilities for classifying data at least at the file level of

granularity. If the organization needs to maintain close scrutiny on the flow of data to, from and through its systems, this method is critical. It will remain in the final framework.

Technical review of operating system changes has resulted in a low cost/effectiveness ratio, but it's necessity in maintaining legitimate system security cannot be disputed. It will remain in the final framework.

Bill back systems are not generally used in commercial organizations today. Results have indicated that most respondents agree that its implementation can result in a significantly more secure system. The perceived cost/effectiveness ratio is also quite high, being the least cost/effective measure evaluated. Despite this, it's use as a diagnostic tool in determining the occurrence of computer abuse has been proven, and it is one of the few tools available to the security professional that can be used to monitor past system resource consumption levels on a system, process or user basis. Despite it's lack of implementation in most commercial organizations today, operating system facilities or third party solutions are available to implement a bill back system on nearly any computer system. I recommend that it remain as part of the final framework.

The Bell-Lapadula data rule has resulted in very high scores in all tests, but this is from only 2 respondents familiar with the method. Its highly technical nature, coupled with it's relative obscurity and difficulty in implementation, make it less desirable as a

security tool. It is my recommendation that it be removed from the final framework, and a specialized database management system (DBMS) be purchased to implement this method should it prove necessary.

Further research possibilities include the validation of the remainder of the framework, the evaluation of new security methods and their placement in this framework, and questionnaire methods that can be used to increase the response rate in security sensitive areas.

It is important to note that the previous conclusions are based on the results of the statistical analysis conducted using the questionnaire responses received. The low response rate makes it difficult to assert any finality to these conclusions. Future research in this area should include the confirmation of these conclusions

Bibliography

Blanton, J. Ellis & Rosenberger (1991), Joann, Determining Your Information System's Vulnerability to Viruses, Journal of Systems Management, pp 10-27

Boockholdt, J.L, (1989), Implementing Security and Integrity in Micro-Mainframe Networks, MIS Quarterly, June, pp 135-144

Chalmers, Leslie S. (1986), An Analysis of the Differences Between the Computer Security Practices in the Military and Private Sectors, Proceedings of the 1986 Symposium on Security and Privacy , IEEE Computer Society Press, pp 71-74

Clark, David D. and Wilson, David R.(1987), A Comparison of Commercial and Military Computer Security Policies, Proceedings of the 1987 Symposium on Security and Privacy , IEEE Computer Society Press, pp 184-194

Cook, Mark G. (1988), Unravelling PC Data-Security Confusion, Internal Auditor, December, pp 49-53

Courtney, Robert H. Jr.,(1986), Security Measures are Inherently Undesirable, EDPACS, March, pp 9-12

CSSC (Canadian System Security Centre) (1993), The Canadian Trusted Computer Product Evaluation Criteria, Version 3.0e

Dektronix Inc. (1995), JANUS Internet Firewall Server - Frequently Asked Questions,

DOD (1983), Department of Defense: Trusted Computer System Evaluation Criteria, 15 Aug 1983, CSC-STD-001-83, AKA "The Orange Book"

Donovan, Steve, (1993), Security of PCs in a Distributed Environment, Computers & Security, Vol 12, pp 28-31

EDPACS,(no author cited)(1978), Potential Areas of Concern in EDP Risk Analysis, August, pp 6-8

Factory Mutual System (1993), Loss Prevention Data for Electronic Data Processing Systems, Factory Mutual Engineering Corp.

Fahn, Paul (1993), Answers to Frequently Asked Questions About Today's Cryptography, RSA Laboratories, Version 2.0, draft 2f

Farmer, Dan & Venema, Wietse (1993), Improving The Security of Your Site by Breaking Into It, retrieved from USENET SIG: comp.security.unix

Fine, Leonard H. (1978), The Total Computer Security Concept, EDPACS, Volume V, No. 11, May, pp 1-9

Fine, Leonard H. (1982), The Total Computer Security Concept and Security Policy, EDPACS, Volume X, No. 5, pp 1-20

Fuentes, Tom (1997), Q&A: Unreported Computer Crime, Ziff-Davis TV Inc., May 12th,1997, http://www.thesite.com/0597w3/life/life550jump4_051297.html

Gibson, Micheal L. & Hughes, Cary T.(1994), Systems Analysis and Design: A Comprehensive Methodology with Case, Boyd & Fraser Publishing Company

Goyal, M.L. & Singh, G.V., (1991), Access Control In Distributed Heterogeneous Database Management, Computers and Security, Vol. 10, No. 7, pp 661-669

Hoffer, Jeffrey A. & Straub, Detmar W. Jr.,(1989), The 9 to 5 Underground: Are you Policing Computer Crimes?, Sloan Management Review, Summer, pp 35-42

Jamieson, Rodger & Low, Graham (1989), Security and Control Issues in Local Area Network Design, Computers and Security, Issue 8, pp 305-316

Kuris, Ron (1994), Frequently Asked Questions about SOCKS,rk@unify.com

Laferriere, Claude, (1990), A Discussion of Implementation Strategies for Secure Database Management Systems, Computers and Security, Vol. 9, No. 3, pp 235-244

Laudon, Kenneth C. & Laudon, Jane Price (1988), Management Information Systems: A Contemporary Perspective, Macmillan Publishing Company

Lee, John A., Segal, Gerald & Steier, Rosalie,(1986) Positive Alternatives: A Report On An ACM Panel on Hacking, Communications of the ACM, Vol. 29, No. 4, pp 297-299

Loch, Karen D., Carr, Houston H. & Warkentin, Merrill E. (1992), Threats to Information Systems: Today's Reality, Yesterday's Understanding, MIS Quarterly, June, pp 173-186

Mace, Scott,(1989), Distributed Data: Sizing up the Challenge, Infoworld, July 31st, 1989, p.39

McHugh, John & Thuraisingham, Bhavani M., (1988), Multilevel Security Issues in Distributed Database Management Systems, Computers and Security, Vol. 7, No. 4, pp 387-395

Mylott, Thomas R. III, (1985), Computer Security and The Threats from Within, The Office, March, pp 45,46,190

NCSC (1988), Glossary of Computer Security Terms, NCSC-TG-004

NCSC (1992), A Guide to Understanding Object Reuse in Trusted Systems, NCSC-TG-018, Version 1

Parker, Donn B. (1982), Computer Crime: Computer Security Techniques, prepared for the US Department Of Justice by SRI International, Grant No. 80-BJ-CX-0015

Parker, Donn B. (1975), Computer Abuse Perpatrators and Vulnerabilities of Computer Systems, Stanford Research Institute, SRI Project 5068

Perry, William E. & Kuong, Javier F. (1981), EDP Risk Analysis and Controls Justification, Management Advisory Publication

Ranum, Marcus J.(1994), Firewalls-FAQ:Frequently Asked Questions about Firewalls, retrieved from mailing list on firewalls: firewalls@greatcircle.com

Rapalus, Patrice (1997), Computer crime continues to increase, Reported losses total over $100 million, Computer Security Institute, March 6, 1997

Ruder, Brian & Madden, J.D,(1978) Computer Science and Technology: An Analysis of Computer Security Safeguards for Detecting and Preventing Intentional Computer Misuse, prepared for the US Department Of Commerce

Sterling, Bruce (1992), The Hacker Crackdown: Law and Disorder on the Electronic Frontier, Bantam Books

Straub, Detmar W. Jr. & Nance, William D., (1990), Discovering and Disciplining Computer Abuse in Organizations: A Field Study, MIS Quarterly, March, pp 45-60

Straub, Detmar W. Jr., (1990), Effective IS Security: An Empirical Study, Information Systems Research, (1:3), p CE5-276

Thuraisingham, Bhavani M., & Rubinovitz, Harvey H., (1992), Multilevel Security Issues in Distributed Database Management Systems-III, Computers and Security, Vol. 11, No. 7, pp 661-674

Thuraisingham, Bhavani M., (1991), Multilevel Security Issues in Distributed Database Management Systems II, Computers and Security, Vol. 10, No. 8, pp 727-747

von Solms, R. van de Haar, H., von Solms, S.H., and Caelli, W.J. (1994), A framework for information security evaluation, Information & Management, Vol 26, pp 143-153

Vaughn, R.B Jr., Saiedan, H. and Unger E.A. (1993), A Questionnaire of Security Issues in Office Computation and the Application of Secure Computer Models to Office Systems, Computers and Security, Vol 12, No. 1, pp 79-97

Whitten, Jeffrey l, Bentley, Lonnie D. & Barlow, Victor M.(1989), Systems Analysis and Design Methods, 2nd Edition, Irwin, 1989

Wong, Ken (1986), Effective Computer Security Management, EDPACS, July, pp 7-11

Wong, Ken (1987), Security and Office Automation, EDPACS, September, pp 5-11

Appendix A:

Questionnaire and Demographic Information of the Sample

Demographic Breakdown of Questionnaire Recipients	
Province	**Number**
Quebec	85
Ontario	162
Manitoba	40
Saskatchewan	27
Alberta	121
Vancouver	46
Yukon	4
Total	485

Note: The following questionnaire is identical to the questionnaire sent to the sample, with one exception; the question identification numbers (QIN) located within parentheses after each question were not present in the original questionnaire.

Dear Sir/Madam,

As a professional in the information systems field, you may be aware of the lack of commercially viable computer security frameworks. Those that currently exist have proven to be expensive and complicated to use for standard commercial organizations.

The purpose of this questionnaire is to validate a commercially viable computer security implementation framework which, when completed, will allow commercial organizations to operate securely in a networked environment. This framework will take effectiveness and cost into account, allowing organizations to design security using a cost/benefit scenario.

The questionnaire will take approximately 30 minutes of your time to complete. We would greatly appreciate it if you could complete the questionnaire and return it in the enclosed self-addressed, stamped envelope as soon as possible.

The questionnaire is anonymous. The respondents cannot be identified. In addition, the individual responses will be kept confidential, and only aggregate data will be published.

Thank You

Affordable computer security is an important issue to many organizations. We would like to thank the professionals who answer this questionnaire. It is only through your expertise that we can develop and validate a model that may prove useful to many organizations in the future.

Sincerely,

Marc Bouffard

Encl.

If you want a copy of the refined questionnaire along with a summary of the analysis for personal use, please indicate your address in the space provided below.

_____

_____

_____

Questionnaire Instructions

The questions in this survey focus on only one aspect of computer security: **system security**.

Many concepts are used, some of which may be unfamiliar. In order to asssure that you have enough information to answer the questions, a few concepts are explained.

System Development Life Cycle (SDLC)

For simplicity, we have settled on the following 4 step systems development life cycle: system analysis, system design, system implementation and system maintenance (or support). This reduces the complexity of the SDLC, limiting it to the minimum number of steps feasible in systems development, while allowing the timing issues needed for the framework to be addressed.

1. System Analysis is the study of a current business system and its problems, the definition of business needs and requirements, and the evaluation of alternative solutions.

2. Systems Design is the general and detailed specification of a computer-based solution that was selected during systems analysis. This includes hardware and software purchasing or development decisions.

3. Systems implementation is placing the system into operation. Computer programs are written and tested, managers and users are trained to use the new system, and operations are converted to the new system.

4. Systems maintenance (or support) is the ongoing support of the system after it has been placed into operation. This includes program maintenance and system improvement.

These stages exist in most methodologies, though not necessarily in the order defined by the SDLC. They describe the necessary steps that will be completed when developing any information system.

Computer System

In order to answer the following question, you must have a specific computer system in mind. This system must have the following attributes:

- it must be networked (inhouse or connected to an external network such as the Internet)
- it must support more than one user
- it may be composed of any combination of hardware or software

Firewalls

Dedicated hardware and software to prevent unauthorized actions, used exclusively in network environments.

Section I:

A- Please rate your familiarity with firewalls by circling the appropriate number, where 1 corresponds to very unfamiliar and and 5 corresponds to very familiar.

Very Unfamiliar 1 2 3 4 5 Very Familiar (1)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of firewalls. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (2)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (3)

C- Please indicate to what extent you agree or disagree with the following procedures for developing firewalls at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: When the system under development will be connected to an untrusted network, especially the Internet, then the use of a firewall will be considered..... (4)

When the system is deemed very sensitive to the organization, then the network connection is not attempted..... (5)

When a firewall is required, any operations that need to be available between the system (or network) and the untrusted network are determined, and the firewall is designed to allow these operations..... (6)

Design: When a network connection is deemed necessary, then the firewall is designed prior to the system..... (7)

Implementation: Once the system under design is connected to the firewall, the firewall is evaluated to assure that it provides appropriate security to the system..... (8)

If the system remains vulnerable, then it should be disconnected from the untrusted network immediately (9)

Maintenance: To ensure that required security levels between the untrusted network and the organizations system are being maintained, the firewall must be audited regularly..... (10)

D- When developing a firewall for your organization, to what extent were the above tasks followed? Indicate N/A if a firewall was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (11)

Remote Terminal Physical Security

Assuring that remote terminals are valid and secure.

Section I:

A- Please rate your familiarity with remote terminal physical security.

Very Unfamiliar 1 2 3 4 5 Very Familiar (12)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of remote terminal physical security. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (13)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (14)

C- Please indicate to what extent you agree or disagree with the following procedures for developing remote terminal physical security at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: When it is not necessary, or when the data on the system is of a sufficiently sensitive nature, then access from external sources will be disallowed..... (15)

Design: When external communications are deemed necessary, then facilities will be put in place to assure the security of these connections..... (16)

Implementation: When the system is implemented, the external communications lines will be evaluated to assure secure and effective..... (17)

Maintenance: The necessity of external communications will be periodically re-evaluated to determine their continued necessity..... (18)

Security and the availability of alternate channels will be tested to assure their continued functionality..... (19)

D- When developing remote terminal physical security for your organization, to what extent were the above tasks followed? Indicate N/A if remote terminal physical security was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (20)

Restricted Use of System Utility Programs

Allowing access to system utility programs only to authorized users.

Section I:

A- Please rate your familiarity with restricting the use of system utility programs.

Very Unfamiliar 1 2 3 4 5 Very Familiar (21)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of restricting the use of system utility programs. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (22)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (23)

C- Please indicate to what extent you agree or disagree with the following procedures for introducing restrictions on the use of system utility programs at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: Appropriate controls are placed on all utility programs to allow them to be accessed only by authorized personnel..... (24)

Maintenance: The entire directory structure is occasionally searched for any "home-made" utility ..... (25)

Utility program logs are maintained and scanned to assure that unauthorized access is being prevented... (26)

D- When restricting the use of utility programs in your organization, to what extent were the above tasks followed? Indicate N/A if restricting system utility programs was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (27)

Assign Files and Programs to Users

Responsibility for the availability and maintenance of files and programs should be attributed to the user, when appropriate.

Section I:

A- Please rate your familiarity with assigning files and programs to users.

Very Unfamiliar 1 2 3 4 5 Very Familiar (28)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of assigning files and programs to users. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (29)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (30)

C- Please indicate to what extent you agree or disagree with the following procedures for assigning files and program to users at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: The employees that will have access to the system are classified into groups by job function..(31)

Design: Directories are seperated by job functions to isolate files and programs from unauthorized access.....(32)

Implementation: The system is implemented and tested by carefully monitoring initial system use.... (33)

Privacy and security must be maintained. but not at the expense of user productivity.....(34)

Maintenance: As new data and programs are assigned to the system, the permissions involved are carefully considered and monitored.....(35)

User productivity is continuously monitored to assure that security is no more counter-productive then it needs to be.....(36)

D- When assigning files and programs to users within your organization, to what extent were the above tasks followed? Indicate N/A if a programs and files were not assigned to users.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (37)

Data Classification

Access labels should be attached to each piece of data (file, record, field) depending on the granularity of the operating or database management system.

Section I:

A- Please rate your familiarity with data classification.

Very Unfamiliar 1 2 3 4 5 Very Familiar (38)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of data classification. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (39)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (40)

C- Please indicate to what extent you agree or disagree with the following procedures for data classification at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: The data necessities of each job function are ascertained..... (41)

Design: The granularity necessary to implement an appropriately secure system is determined..... (42)

Once the necessary garanularity is determined, the scheme used to manage access labels is determined (multi or uni-level? OS or DBMS ?)..... (43)

Implementation: The system is thoroughly tested to assure that access labels are being properly managed and effectively implemented..... (44)

Data levels must remain intact even if the data is moved from one machine in the network to another. (45)

Maintenance: Access labels to new data are ascertained and implemented..... (46)

Existing access labels are regularly audited to assure their continued appropriateness..... (47)

D- When classifying data for your organization, to what extent were the above tasks followed? Indicate N/A if data classification was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (48)

Bell-Lapadula Data Rule

Use of the simple and *-properties when determining read or write access to objects.

Section I:

A- Please rate your familiarity with the Bell-Lapadula data rule.

Very Unfamiliar 1 2 3 4 5 Very Familiar (49)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of the Bell-Lapadula data rule. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (50)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (51)

C- Please indicate to what extent you agree or disagree with the following procedures for developing the Bell-Lapadula data rule at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Design: The necessity of implementing the Bell-Lapadula data rule is ascertained..... (52)

If this data rule is needed, then a DBMS which allows the application of this rule is acquired..... (53)

D- When developing the Bell-Lapadula data rule for use in your organization, to what extent were the above tasks followed? Indicate N/A if the Bell-Lapadula data rule was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (54)

Technical Review of Operating System Changes

Each change to the operating system should lead to a detailed technical review of the changes, and their impact to the organization.

Section I:

A- Please rate your familiarity with technical reviews of OS changes.

Very Unfamiliar 1 2 3 4 5 Very Familiar (55)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of technically reviewing OS changes. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (56)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (57)

C- Please indicate to what extent you agree or disagree with the following procedures for managing the technical review of operating system changes at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Maintenance: Changes to the operating system or system work environment are carefully monitored to assure that they do not compromise the following areas: security, worker productivity, application performance and system integrity..... (58)

D- When technically reviewing OS changes within your organization, to what extent were the above tasks followed? Indicate N/A if technical review of OS changes is not part of your security precautions.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (59)

Cryptographic Protection

Cryptography to assure data security on local machines and communication lines.

Section I:

A- Please rate your familiarity with cryptographic protection.

Very Unfamiliar 1 2 3 4 5 Very Familiar (60)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of cryptographic protection. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (61)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (62)

C- Please indicate to what extent you agree or disagree with the following procedures for introducing cryptographic protection at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: The sensitivity levels of the data are determined..... (63)

Design: If it is determined that cryptographic measures are necessary, then their function is ascertained (eg. remote encryption, encryption for transport, communication encryption, encryption of data on the system, email encryption or network authentication)..... (64)

If one encryption scheme can be used to fulfil all encryption requirements, then one method is chosen and used throughout the system..... (65)

Implementation: Once encryption tool are implemented, their use must be carefully monitored..... (66)

Maintenance: New cryptographic techniques should be monitored for possible adoption..... (67)

Legislative issues should also be monitored to assure that the use of cryptographic tools by the organization does not contradict any laws..... (68)

D- When developing a cryptographic protection scheme for your organization, to what extent were the above tasks followed? Indicate N/A if cryptographic protection was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (69)

User Authentication

Assuring the accuracy of user identification for proper security management.

Section I:

A- Please rate your familiarity with user authentication schemes.

Very Unfamiliar 1 2 3 4 5 Very Familiar (70)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of user authentication. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (71)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (72)

C- Please indicate to what extent you agree or disagree with the following procedures for introducing user authentication schemes at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Analysis: The extent of user authentication needed for the system is determined..... (73)

Design: When high levels of technology are to be used (keycards, retinal scan etc.) for user authentication then a systematic review of all the technologies available is conducted to find the one most suited..... (74)

When a password scheme is selected, a procedure is implemented to verify the passwords and assure that users have not chosen a password that will be too easy to guess..... (75)

Audit logs for failed logins are established so that attempts to "hack" into a users account can be identified..... (76)

Implementation:

Users are instructed to change their default passwords as soon as the system comes online..... (77)

Technological user authentication schemes are carefully explained to users, along with any precautions that should be taken to avoid any unnecessary risk (eg. report lost or stolen keycards)..... (78)

Maintenance: Procedures are established to ensure that any security precautions taken remain effective..... (79)

D- When developing a user authentication scheme for your organization, to what extent were the above tasks followed? Indicate N/A if user authentication was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (80)

Automatic, Timed Terminal Logoff

Automatic terminal log-offs after a certain period of inactivity.

Section I:

A- Please rate your familiarity with automatic, timed terminal logoff.

Very Unfamiliar 1 2 3 4 5 Very Familiar (81)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of automatic, timed terminal logoff. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (82)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (83)

C- Please indicate to what extent you agree or disagree with the following procedures for implementing automatic, timed terminal logoff at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Design: Software is written or purchased to support this feature if it does not already exists within the operating system..... (84)

Implementation: This facility is activated within the operating system. Programs which were written or purchased must be implemented and tested..... (85)

D- When developing automatic, timed terminal logoff for your organization, to what extent were the above tasks followed? Indicate N/A if a automatic, timed terminal logoff was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (86)

Bill Back System

Billing back computer resource expenses to users. Helps identify unusual resource use.

Section I:

A- Please rate your familiarity with bill back systems.

Very Unfamiliar 1 2 3 4 5 Very Familiar (87)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of a bill back system. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (88)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (89)

C- Please indicate to what extent you agree or disagree with the following procedures for developing a bill back system at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Design: If the operating system does not provide the facilities for a billback system, then software is written inhouse or purchased..... (90)

Implementation: The system is tested online. The overhead necessary to manage such a system must not significantly impair system performance..... (91)

Maintenance: Periodically, the billback system logs is carefully scanned to determine regular resource consumption patterns. Any deviation from these patterns is closely scrutinized to determine the cause. (92)

D- When developing a bill back system for your organization, to what extent were the above tasks followed? Indicate N/A if a bill back system was not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (93)

Hardware Monitors

Monitoring hardware usage throughout the organization for unusual usage levels.

Section I:

A- Please rate your familiarity with hardware monitors.

Very Unfamiliar 1 2 3 4 5 Very Familiar (94)

If you indicated 2 or less on this question, please proceed to the measure on the next page.

B- Please rate the effectiveness and cost of hardware monitors. Effectiveness is rated from 1 to 5 where 1 is very ineffective and 5 is very effective. Cost is rated from 1 to 5 where 1 is very inexpensive and 5 is very expensive.

Effectiveness: Very Ineffective 1 2 3 4 5 Very Effective (95)

Cost: Very Inexpensive 1 2 3 4 5 Very Expensive (96)

C- Please indicate to what extent you agree or disagree with the following procedures for monitoring hardware at each phase of the system development life cycle listed below. The scale is from 1 to 5 where 1 indicates complete disagreement, 3 indicates no opinion and 5 indicates completely agreement. Indicate your selection in the blank following each question.

Scale: Completely Disagree 1 2 3 4 5 Completely Agree

Design: If the hardware does not provide the facilities for monitoring itself, then specialized hardware is purchased to perform this function..... (97)

Implementation: The system is tested online. The overhead necessary to manage such a system must not significantly impair system performance..... (98)

Maintenance: Periodically, the hardware monitor logs are carefully scanned to determine regular resource consumption patterns. Any deviation from these patterns should be closely scrutinized to determine the cause..... (99)

D- When developing hardware monitors for your organization, to what extent were the above tasks followed? Indicate N/A if a hardware monitors are not implemented.

Not Followed at All 1 2 3 4 5 Followed Completely N/A (100)

Section II: Security Evaluation

Please rate the overall effectiveness of the security measures currently implemented on your computer system.

Very Ineffective 1 2 3 4 5 Very Effective (101)

Section III: Organization

A- How many employees work in your organization?

Less than 10 **10-100** **101-500** **501-1000** **More than 1000** (102)

B- How many users use your computer system?

Less than 10 **10-50** **51-150** **151-500** **More than 500** (103)

C- Were you involved in the development of this system? **Yes** **No** (104)

D- How was this system developed? **Inhouse** **Purchased** (105)

If this system was developed inhouse, please indicate the methodology used in its development.

1- Systems Development Life Cycle (SDLC) (106)
2- Prototyping
3- Computer Aided Systems Engineering (CASE)
4- Any combination of the above
5- None of the above

E- Who are the main users of the system? **Employees** **Clients** **Third Party** (107)

Appendix B:

Chi-Square Generation Program and Chi-Square Tables

(written in Qbasic)

```
DECLARE SUB PRTCHI ()
DECLARE SUB GENCELL ()
DECLARE SUB GENCHI ()
DIM SHARED CELL(2, 2)
DIM SHARED CELLTOT(2, 2)
DIM SHARED ROWTOT(2)
DIM SHARED COLTOT(2)
DIM SHARED N1, N2, CN1, CN2, CHITOT(5000) AS SINGLE
DIM SHARED CHIOCC(5000, 2) AS SINGLE
DIM CHISORT, PER, PER1, CUMPER, CUMPER1 AS SINGLE
DIM SHARED OBS, A, B, ITE, VALCHI
CLS
INPUT "Enter number of observations:"; OBS
INPUT "Enter number of iterations:"; ITE
PRINT "Enter number of output matrix for validation (1-"; ITE; ", 0 for none):";
INPUT VALCHI
IF VALCHI <> 0 THEN
   INPUT "Please make sure your printer is ready... <Any key>"; GARB
END IF

OPEN "CHI2" FOR OUTPUT AS #1
PRINT #1, "         ===== Sumlated Chi Square Distribution for ";OBS; "
Observations====="
PRINT #1, ""
PRINT #1, "Number of observations:"; OBS
PRINT #1, "Number of iterations:"; ITE
PRINT #1, ""
PRINT #1, "Chi-Square   Obs.            (1-P)      Chi-Square   Obs.            (1-P)"
PRINT #1, "-------------------------------------------------------------------------------"
TOTITE = ITE

REM #### Generate chi squared values
FOR B = 1 TO ITE
REM ### Determine cell probabilities for matrix
RANDOMIZE TIMER
N1 = RND
N2 = RND
CN1 = 1 - N1
CN2 = 1 - N2
CALLGENCELL:
ERASE CELLTOT
FOR A = 1 TO OBS
   GENCELL
   FOR Q = 1 TO 2
     FOR R = 1 TO 2
       CELLTOT(Q, R) = CELLTOT(Q, R) + CELL(Q, R)
     NEXT R
   NEXT Q
NEXT A
ROWTOT(1) = CELLTOT(1, 1) + CELLTOT(2, 1)
ROWTOT(2) = CELLTOT(1, 2) + CELLTOT(2, 2)
COLTOT(1) = CELLTOT(1, 1) + CELLTOT(1, 2)
COLTOT(2) = CELLTOT(2, 1) + CELLTOT(2, 2)
IF ROWTOT(1) = 0 OR ROWTOT(2) = 0 OR COLTOT(1) = 0 OR COLTOT(2) = 0 THEN
  GOTO CALLGENCELL
END IF
MATTOT = ROWTOT(1) + ROWTOT(2)
GENCHI
IF VALCHI = B THEN
  PRTCHI
END IF

REM ### Initialize Celltotal
ERASE CELLTOT

NEXT B
REM #### Sort Chi Values
SORTCHI:
SORT = 0
FOR W = 2 TO ITE
   IF CHITOT(W) < CHITOT(W - 1) THEN
```

```
      CHISORT = CHITOT(W)
      CHITOT(W) = CHITOT(W - 1)
      CHITOT(W - 1) = CHISORT
      SORT = 1
   END IF
NEXT W
IF SORT = 1 THEN
   GOTO SORTCHI
END IF



ERASE CHIOCC
REM #### Initialize the first element in the chi squared occurence array
CURR = 1
CHIOCC(CURR, 1) = CHITOT(W)
CHIOCC(CURR, 2) = 1
FOR W = 2 TO ITE
   IF CHITOT(W) = CHIOCC(CURR, 1) THEN
      CHIOCC(CURR, 2) = CHIOCC(CURR, 2) + 1
   ELSE
      CURR = CURR + 1
      CHIOCC(CURR, 1) = CHITOT(W)
      CHIOCC(CURR, 2) = 1
   END IF
NEXT W



REM ### Print chi squared results
CUMPER = 0
CUMPER1 = 0
TOTOCC = 0
FOR QQ = 1 TO CURR STEP 2
   PER = (CHIOCC(QQ, 2) / TOTITE) * 100
   PER1 = (CHIOCC(QQ + 1, 2) / TOTITE) * 100
   CUMPER = CUMPER1 + PER
   CUMPER1 = CUMPER + PER1
   PRINT #1, USING "###.##### #####.#### ###.##% ###.##% | ###.##### #####.#### ###.##%
###.##%"; CHIOCC(QQ, 1); CHIOCC(QQ, 2); PER; CUMPER; CHIOCC(QQ + 1, 1); CHIOCC(QQ + 1,
2); PER1; CUMPER1
   TOTOCC = TOTOCC + CHIOCC(QQ, 2) + CHIOCC(QQ + 1, 2)
NEXT QQ
CLOSE #1

SUB GENCELL
REM #### Generate 2X2 Matrix.
DIM N3 AS SINGLE
RANDOMIZE TIMER
N3 = RND
ERASE CELL
REM #### Assign value to one of 4 cells
IF N3 <= (N1 * N2) THEN
         CELL(1, 1) = 1
ELSEIF N3 <= ((CN1 * N2) + (N1 * N2)) AND N3 > (N1 * N2) THEN
         CELL(2, 1) = 1
ELSEIF N3 <= ((N1 * N2) + (CN1 * N2) + (N1 * CN2)) AND N3 > ((CN1 * N2) + (N1 * N2)) THEN
         CELL(1, 2) = 1
ELSEIF N3 > ((N1 * N2) + (CN1 * N2) + (N1 * CN2)) THEN
         CELL(2, 2) = 1
END IF
END SUB

SUB GENCHI
REM #### Determine chi squared values from pre-generated 2X2 matrix
DIM CHITEMP AS SINGLE
DIM EXPEC AS SINGLE
DIM ROUNDER AS LONG
DIM CHIHOLD AS DOUBLE
CHITOT = 0
CHIHOLD = 0
```

```
FOR X = 1 TO 2
     FOR Y = 1 TO 2
       OBSVAL = CELLTOT(X, Y)
       EXPEC = ((ROWTOT(Y) * COLTOT(X)) / OBS)
       CHITEMP = (((OBSVAL - EXPEC) ^ 2) / EXPEC)
       CHIHOLD = CHIHOLD + CHITEMP
     NEXT Y
NEXT X
ROUNDER = CHIHOLD * 10000
CHITOT(B) = ROUNDER / 10000
END SUB

SUB PRTCHI
   PRINT "Printing sample "; B; "..."
   LPRINT ""
   LPRINT "Chi Square Sample"
   LPRINT "-----------------"
   LPRINT "Iteration: "; B
   LPRINT "Total Observations: "; OBS
   LPRINT ""
   LPRINT "┌─────────┬─────────┐"
   LPRINT "│ "; CELLTOT(1, 1), "│ "; CELLTOT(2, 1), "│ "; ROWTOT(1)
   LPRINT "├─────────┼─────────┤"
   LPRINT "│ "; CELLTOT(1, 2), "│ "; CELLTOT(2, 2), "│ "; ROWTOT(2)
   LPRINT "└─────────┴─────────┘"
   LPRINT "  "; COLTOT(1), "  "; COLTOT(2), OBS

   LPRINT "Chi Square: "; CHITOT(B)
   LPRINT CHR$(12)
END SUB
```

===== Simulated Chi Square Distribution for 15 Observations =====

Number of observations: 15
Number of iterations: 1000

Chi-Square	Obs.		(1-P)	Chi-Square	Obs.		(1-P)
0.00000	33.0000	3.30%	3.30%	0.00770	13.0000	1.30%	4.60%
0.01030	26.0000	2.60%	7.20%	0.02440	10.0000	1.00%	8.20%
0.04460	9.0000	0.90%	9.10%	0.06940	20.0000	2.00%	11.10%
0.07650	45.0000	4.50%	15.60%	0.08520	19.0000	1.90%	17.50%
0.09620	22.0000	2.20%	19.70%	0.13390	16.0000	1.60%	21.30%
0.15000	8.0000	0.80%	22.10%	0.16480	49.0000	4.90%	27.00%
0.17050	18.0000	1.80%	28.80%	0.18520	5.0000	0.50%	29.30%
0.22730	10.0000	1.00%	30.30%	0.26790	56.0000	5.60%	35.90%
0.28850	18.0000	1.80%	37.70%	0.35500	14.0000	1.40%	39.10%
0.38960	34.0000	3.40%	42.50%	0.41670	12.0000	1.20%	43.70%
0.51140	11.0000	1.10%	44.80%	0.53570	50.0000	5.00%	49.80%
0.57690	21.0000	2.10%	51.90%	0.57880	8.0000	0.80%	52.70%
0.60000	8.0000	0.80%	53.50%	0.60270	9.0000	0.90%	54.40%
0.64250	15.0000	1.50%	55.90%	0.68180	11.0000	1.10%	57.00%
0.71430	38.0000	3.80%	60.80%	0.83920	14.0000	1.40%	62.20%
0.93750	42.0000	4.20%	66.40%	1.02880	9.0000	0.90%	67.30%
1.11110	6.0000	0.60%	67.90%	1.15380	22.0000	2.20%	70.10%
1.22450	26.0000	2.60%	72.70%	1.25000	9.0000	0.90%	73.60%
1.29810	11.0000	1.10%	74.70%	1.36360	6.0000	0.60%	75.30%
1.51860	6.0000	0.60%	75.90%	1.53850	14.0000	1.40%	77.30%
1.60710	27.0000	2.70%	80.00%	1.72670	4.0000	0.40%	80.40%
1.75930	9.0000	0.90%	81.30%	1.87500	13.0000	1.30%	82.60%
1.98350	1.0000	0.10%	82.70%	2.01920	10.0000	1.00%	83.70%
2.14290	29.0000	2.90%	86.60%	2.26850	2.0000	0.20%	86.80%
2.40000	3.0000	0.30%	87.10%	2.50000	6.0000	0.60%	87.70%
2.63740	7.0000	0.70%	88.40%	2.68490	9.0000	0.90%	89.30%
2.72730	4.0000	0.40%	89.70%	2.78410	5.0000	0.50%	90.20%
2.94640	16.0000	1.60%	91.80%	2.96300	2.0000	0.20%	92.00%
3.06820	5.0000	0.50%	92.50%	3.23340	4.0000	0.40%	92.90%
3.28120	5.0000	0.50%	93.40%	3.34820	3.0000	0.30%	93.70%
3.46150	6.0000	0.60%	94.30%	3.61610	2.0000	0.20%	94.50%
3.63640	6.0000	0.60%	95.10%	4.26140	1.0000	0.10%	95.20%
4.28570	11.0000	1.10%	96.30%	4.61540	3.0000	0.30%	96.60%
4.77270	3.0000	0.30%	96.90%	5.00000	5.0000	0.50%	97.40%
5.10420	2.0000	0.20%	97.60%	5.40180	2.0000	0.20%	97.80%
5.62500	3.0000	0.30%	98.10%	6.23380	1.0000	0.10%	98.20%
6.34620	2.0000	0.20%	98.40%	6.96430	6.0000	0.60%	99.00%
7.35000	1.0000	0.10%	99.10%	7.50000	1.0000	0.10%	99.20%
9.23080	4.0000	0.40%	99.60%	15.00000	4.0000	0.40%	100.00%

===== Simulated Chi Square Distribution for 21 Observations =====

Number of observations: 21
Number of iterations: 1000

Chi-Square	Obs.		(1-P)	Chi-Square	Obs.		(1-P)
0.00028	3.0000	0.30%	0.30%	0.00046	4.0000	0.40%	0.70%
0.00074	2.0000	0.20%	0.90%	0.00092	1.0000	0.10%	1.00%
0.00286	2.0000	0.20%	1.20%	0.00293	2.0000	0.20%	1.40%
0.00298	8.0000	0.80%	2.20%	0.00397	12.0000	1.20%	3.40%
0.00529	5.0000	0.50%	3.90%	0.00533	9.0000	0.90%	4.80%
0.00586	11.0000	1.10%	5.90%	0.00649	4.0000	0.40%	6.30%
0.00714	6.0000	0.60%	6.90%	0.00762	1.0000	0.10%	7.00%
0.00866	9.0000	0.90%	7.90%	0.00879	2.0000	0.20%	8.10%
0.00952	15.0000	1.50%	9.60%	0.00984	10.0000	1.00%	10.60%
0.01042	1.0000	0.10%	10.70%	0.01082	18.0000	1.80%	12.50%
0.01190	10.0000	1.00%	13.50%	0.01282	4.0000	0.40%	13.90%
0.01299	13.0000	1.30%	15.20%	0.01429	19.0000	1.90%	17.10%
0.01515	6.0000	0.60%	17.70%	0.01667	12.0000	1.20%	18.90%
0.01732	18.0000	1.80%	20.70%	0.01905	19.0000	1.90%	22.60%
0.02083	5.0000	0.50%	23.10%	0.02143	3.0000	0.30%	23.40%
0.02198	1.0000	0.10%	23.50%	0.02381	1.0000	0.10%	23.60%
0.02473	8.0000	0.80%	24.40%	0.02564	4.0000	0.40%	24.80%
0.02607	1.0000	0.10%	24.90%	0.02857	1.0000	0.10%	25.00%
0.03030	3.0000	0.30%	25.30%	0.03297	12.0000	1.20%	26.50%
0.03333	5.0000	0.50%	27.00%	0.03463	1.0000	0.10%	27.10%
0.03571	1.0000	0.10%	27.20%	0.03810	2.0000	0.20%	27.40%
0.04018	12.0000	1.20%	28.60%	0.04167	2.0000	0.20%	28.80%
0.04396	8.0000	0.80%	29.60%	0.04573	1.0000	0.10%	29.70%
0.04762	5.0000	0.50%	30.20%	0.05030	1.0000	0.10%	30.30%
0.05079	2.0000	0.20%	30.50%	0.05128	4.0000	0.40%	30.90%
0.05357	6.0000	0.60%	31.50%	0.05495	4.0000	0.40%	31.90%
0.05541	2.0000	0.20%	32.10%	0.05861	11.0000	1.10%	33.20%
0.06061	6.0000	0.60%	33.80%	0.06667	4.0000	0.40%	34.20%
0.07143	17.0000	1.70%	35.90%	0.08036	1.0000	0.10%	36.00%
0.08117	19.0000	1.90%	37.90%	0.08333	5.0000	0.50%	38.40%
0.08929	17.0000	1.70%	40.10%	0.09351	1.0000	0.10%	40.20%
0.09524	15.0000	1.50%	41.70%	0.09624	12.0000	1.20%	42.90%
0.10000	2.0000	0.20%	43.10%	0.10075	5.0000	0.50%	43.60%
0.10256	3.0000	0.30%	43.90%	0.10286	1.0000	0.10%	44.00%
0.10586	8.0000	0.80%	44.80%	0.10714	5.0000	0.50%	45.30%
0.10823	21.0000	2.10%	47.40%	0.11082	13.0000	1.30%	48.70%
0.11271	3.0000	0.30%	49.00%	0.11905	18.0000	1.80%	50.80%
0.12121	6.0000	0.60%	51.40%	0.12190	9.0000	0.90%	52.30%
0.12500	3.0000	0.30%	52.60%	0.12698	1.0000	0.10%	52.70%
0.13333	5.0000	0.50%	53.20%	0.13638	3.0000	0.30%	53.50%
0.13736	2.0000	0.20%	53.70%	0.14286	14.0000	1.40%	55.10%
0.15639	8.0000	0.80%	55.90%	0.16095	1.0000	0.10%	56.00%
0.16623	1.0000	0.10%	56.10%	0.16667	13.0000	1.30%	57.40%
0.17202	6.0000	0.60%	58.00%	0.17802	1.0000	0.10%	58.10%
0.18286	1.0000	0.10%	58.20%	0.18315	7.0000	0.70%	58.90%
0.19048	10.0000	1.00%	59.90%	0.19680	1.0000	0.10%	60.00%
0.20000	2.0000	0.20%	60.20%	0.20513	2.0000	0.20%	60.40%
0.20813	11.0000	1.10%	61.50%	0.22161	7.0000	0.70%	62.20%
0.22321	1.0000	0.10%	62.30%	0.22894	11.0000	1.10%	63.40%
0.23377	1.0000	0.10%	63.50%	0.24242	5.0000	0.50%	64.00%
0.25000	10.0000	1.00%	65.00%	0.25397	9.0000	0.90%	65.90%
0.25714	4.0000	0.40%	66.30%	0.26299	9.0000	0.90%	67.20%
0.26604	3.0000	0.30%	67.50%	0.26667	3.0000	0.30%	67.80%
0.27473	3.0000	0.30%	68.10%	0.28810	3.0000	0.30%	68.40%
0.28929	8.0000	0.80%	69.20%	0.29264	13.0000	1.30%	70.50%
0.29762	4.0000	0.40%	70.90%	0.31648	3.0000	0.30%	71.20%
0.32051	1.0000	0.10%	71.30%	0.32190	4.0000	0.40%	71.70%
0.33242	1.0000	0.10%	71.80%	0.33333	6.0000	0.60%	72.40%
0.33820	6.0000	0.60%	73.00%	0.33862	2.0000	0.20%	73.20%
0.34286	2.0000	0.20%	73.40%	0.35065	8.0000	0.80%	74.20%
0.36012	2.0000	0.20%	74.40%	0.36630	2.0000	0.20%	74.60%
0.37202	7.0000	0.70%	75.30%	0.37879	1.0000	0.10%	75.40%
0.38571	8.0000	0.80%	76.20%	0.39683	1.0000	0.10%	76.30%
0.40000	1.0000	0.10%	76.40%	0.41026	3.0000	0.30%	76.70%

0.41558	2.0000	0.20%	76.90%	0.41667	4.0000	0.40%	77.30%
0.43896	2.0000	0.20%	77.50%	0.44322	3.0000	0.30%	77.80%
0.44506	1.0000	0.10%	77.90%	0.44643	4.0000	0.40%	78.30%
0.45714	2.0000	0.20%	78.50%	0.46753	8.0000	0.80%	79.30%
0.47365	1.0000	0.10%	79.40%	0.48085	8.0000	0.80%	80.20%
0.48286	1.0000	0.10%	80.30%	0.48485	4.0000	0.40%	80.70%
0.50223	1.0000	0.10%	80.80%	0.50298	1.0000	0.10%	80.90%
0.51429	5.0000	0.50%	81.40%	0.52083	3.0000	0.30%	81.70%
0.52894	3.0000	0.30%	82.00%	0.53333	4.0000	0.40%	82.40%
0.53876	5.0000	0.50%	82.90%	0.54018	3.0000	0.30%	83.20%
0.59264	6.0000	0.60%	83.80%	0.59524	6.0000	0.60%	84.40%
0.60952	1.0000	0.10%	84.50%	0.62338	5.0000	0.50%	85.00%
0.62820	1.0000	0.10%	85.10%	0.62976	1.0000	0.10%	85.20%
0.65190	1.0000	0.10%	85.30%	0.66667	3.0000	0.30%	85.60%
0.67063	5.0000	0.50%	86.10%	0.68571	10.0000	1.00%	87.10%
0.70463	1.0000	0.10%	87.20%	0.72024	6.0000	0.60%	87.80%
0.72321	2.0000	0.20%	88.00%	0.74242	1.0000	0.10%	88.10%
0.75758	2.0000	0.20%	88.30%	0.76190	2.0000	0.20%	88.50%
0.76969	2.0000	0.20%	88.70%	0.77509	1.0000	0.10%	88.80%
0.79121	1.0000	0.10%	88.90%	0.79396	4.0000	0.40%	89.30%
0.80095	2.0000	0.20%	89.50%	0.81667	1.0000	0.10%	89.60%
0.83117	2.0000	0.20%	89.80%	0.83333	3.0000	0.30%	90.10%
0.84661	2.0000	0.20%	90.30%	0.85952	4.0000	0.40%	90.70%
0.86934	3.0000	0.30%	91.00%	0.89286	2.0000	0.20%	91.20%
0.89418	2.0000	0.20%	91.40%	0.91429	1.0000	0.10%	91.50%
0.95628	5.0000	0.50%	92.00%	0.96970	2.0000	0.20%	92.20%
1.00000	3.0000	0.30%	92.50%	1.04167	3.0000	0.30%	92.80%
1.05190	2.0000	0.20%	93.00%	1.10822	2.0000	0.20%	93.20%
1.14502	2.0000	0.20%	93.40%	1.15413	1.0000	0.10%	93.50%
1.17208	1.0000	0.10%	93.60%	1.19048	1.0000	0.10%	93.70%
1.21905	3.0000	0.30%	94.00%	1.25074	1.0000	0.10%	94.10%
1.25952	3.0000	0.30%	94.40%	1.27509	1.0000	0.10%	94.50%
1.28929	1.0000	0.10%	94.60%	1.29018	1.0000	0.10%	94.70%
1.32389	2.0000	0.20%	94.90%	1.33333	2.0000	0.20%	95.10%
1.37574	2.0000	0.20%	95.30%	1.42857	1.0000	0.10%	95.40%
1.43006	1.0000	0.10%	95.50%	1.46104	1.0000	0.10%	95.60%
1.56277	1.0000	0.10%	95.70%	1.58730	1.0000	0.10%	95.80%
1.60714	1.0000	0.10%	95.90%	1.63333	2.0000	0.20%	96.10%
1.71753	1.0000	0.10%	96.20%	1.71905	1.0000	0.10%	96.30%
1.72024	3.0000	0.30%	96.60%	1.88366	1.0000	0.10%	96.70%
1.88929	1.0000	0.10%	96.80%	1.92063	1.0000	0.10%	96.90%
2.00173	1.0000	0.10%	97.00%	2.07202	3.0000	0.30%	97.30%
2.19481	1.0000	0.10%	97.40%	2.56085	1.0000	0.10%	97.50%
2.66667	1.0000	0.10%	97.60%	2.66667	0.0000	0.00%	97.60%

===== Simulated Chi Square Distribution for 25 Observations =====

Number of observations: 25
Number of iterations: 1000

Chi-Square	Obs.		(1-P)	Chi-Square	Obs.		(1-P)
0.00000	5.0000	0.50%	0.50%	0.00110	3.0000	0.30%	0.80%
0.00160	2.0000	0.20%	1.00%	0.00260	4.0000	0.40%	1.40%
0.00280	11.0000	1.10%	2.50%	0.00350	7.0000	0.70%	3.20%
0.00520	1.0000	0.10%	3.30%	0.00640	3.0000	0.30%	3.60%
0.00760	4.0000	0.40%	4.00%	0.01050	7.0000	0.70%	4.70%
0.01150	2.0000	0.20%	4.90%	0.01270	6.0000	0.60%	5.50%
0.01690	1.0000	0.10%	5.60%	0.01890	6.0000	0.60%	6.20%
0.02130	7.0000	0.70%	6.90%	0.02440	4.0000	0.40%	7.30%
0.02670	1.0000	0.10%	7.40%	0.03060	3.0000	0.30%	7.70%
0.03180	4.0000	0.40%	8.10%	0.03310	4.0000	0.40%	8.50%
0.03700	2.0000	0.20%	8.70%	0.04060	5.0000	0.50%	9.20%
0.04340	21.0000	2.10%	11.30%	0.04810	2.0000	0.20%	11.50%
0.05100	1.0000	0.10%	11.60%	0.05250	6.0000	0.60%	12.20%
0.05480	2.0000	0.20%	12.40%	0.06310	4.0000	0.40%	12.80%
0.06960	4.0000	0.40%	13.20%	0.07120	3.0000	0.30%	13.50%
0.07440	6.0000	0.60%	14.10%	0.09060	20.0000	2.00%	16.10%
0.10300	3.0000	0.30%	16.40%	0.10720	4.0000	0.40%	16.80%
0.10820	2.0000	0.20%	17.00%	0.11140	4.0000	0.40%	17.40%
0.11530	2.0000	0.20%	17.60%	0.11570	3.0000	0.30%	17.90%
0.14200	10.0000	1.00%	18.90%	0.14620	6.0000	0.60%	19.50%
0.15740	3.0000	0.30%	19.80%	0.16030	7.0000	0.70%	20.50%
0.16280	8.0000	0.80%	21.30%	0.16350	4.0000	0.40%	21.70%
0.17190	2.0000	0.20%	21.90%	0.18380	6.0000	0.60%	22.50%
0.18490	4.0000	0.40%	22.90%	0.18900	7.0000	0.70%	23.60%
0.19840	32.0000	3.20%	26.80%	0.20170	2.0000	0.20%	27.00%
0.23280	3.0000	0.30%	27.30%	0.23290	5.0000	0.50%	27.80%
0.24350	2.0000	0.20%	28.00%	0.25010	4.0000	0.40%	28.40%
0.26040	17.0000	1.70%	30.10%	0.28700	2.0000	0.20%	30.30%
0.29380	4.0000	0.40%	30.70%	0.29640	12.0000	1.20%	31.90%
0.32160	3.0000	0.30%	32.20%	0.32370	6.0000	0.60%	32.80%
0.32560	2.0000	0.20%	33.00%	0.32890	21.0000	2.10%	35.10%
0.33720	6.0000	0.60%	35.70%	0.36060	8.0000	0.80%	36.50%
0.36450	7.0000	0.70%	37.20%	0.37700	6.0000	0.60%	37.80%
0.37880	15.0000	1.50%	39.30%	0.40510	22.0000	2.20%	41.50%
0.41360	4.0000	0.40%	41.90%	0.41410	10.0000	1.00%	42.90%
0.42740	4.0000	0.40%	43.30%	0.44640	9.0000	0.90%	44.20%
0.46490	2.0000	0.20%	44.40%	0.47590	3.0000	0.30%	44.70%
0.49020	19.0000	1.90%	46.60%	0.50300	2.0000	0.20%	46.80%
0.51970	3.0000	0.30%	47.10%	0.52190	9.0000	0.90%	48.00%
0.52670	5.0000	0.50%	48.50%	0.52910	6.0000	0.60%	49.10%
0.54350	7.0000	0.70%	49.80%	0.58590	15.0000	1.50%	51.30%
0.61790	2.0000	0.20%	51.50%	0.64940	22.0000	2.20%	53.70%
0.67160	1.0000	0.10%	53.80%	0.68040	5.0000	0.50%	54.30%
0.68160	3.0000	0.30%	54.60%	0.68650	13.0000	1.30%	55.90%
0.69440	20.0000	2.00%	57.90%	0.69770	3.0000	0.30%	58.20%
0.70900	9.0000	0.90%	59.10%	0.71080	1.0000	0.10%	59.20%
0.76210	8.0000	0.80%	60.00%	0.80570	2.0000	0.20%	60.20%
0.81850	15.0000	1.50%	61.70%	0.84540	8.0000	0.80%	62.50%
0.85230	9.0000	0.90%	63.40%	0.85300	6.0000	0.60%	64.00%
0.87720	7.0000	0.70%	64.70%	0.88650	4.0000	0.40%	65.10%
0.90700	6.0000	0.60%	65.70%	0.93920	3.0000	0.30%	66.00%
0.96150	10.0000	1.00%	67.00%	0.98720	1.0000	0.10%	67.10%
0.99100	5.0000	0.50%	67.60%	1.00080	2.0000	0.20%	67.80%
1.00920	3.0000	0.30%	68.10%	1.01010	3.0000	0.30%	68.40%
1.02300	2.0000	0.20%	68.60%	1.04170	6.0000	0.60%	69.20%
1.06450	1.0000	0.10%	69.30%	1.06560	1.0000	0.10%	69.40%
1.07660	5.0000	0.50%	69.90%	1.10210	2.0000	0.20%	70.10%
1.10290	4.0000	0.40%	70.50%	1.12850	9.0000	0.90%	71.40%
1.14320	4.0000	0.40%	71.80%	1.15860	2.0000	0.20%	72.00%
1.17550	3.0000	0.30%	72.30%	1.19050	4.0000	0.40%	72.70%
1.21190	2.0000	0.20%	72.90%	1.22280	6.0000	0.60%	73.50%
1.28080	2.0000	0.20%	73.70%	1.32580	9.0000	0.90%	74.60%
1.39080	4.0000	0.40%	75.00%	1.39150	2.0000	0.20%	75.20%
1.41780	1.0000	0.10%	75.30%	1.44930	7.0000	0.70%	76.00%

1.46920	4.0000	0.40%	76.40%	1.47030	2.0000	0.20%	76.60%
1.50380	4.0000	0.40%	77.00%	1.56250	11.0000	1.10%	78.10%
1.60430	4.0000	0.40%	78.50%	1.63410	1.0000	0.10%	78.60%
1.64620	4.0000	0.40%	79.00%	1.70810	2.0000	0.20%	79.20%
1.72360	2.0000	0.20%	79.40%	1.73160	3.0000	0.30%	79.70%
1.75170	2.0000	0.20%	79.90%	1.76480	2.0000	0.20%	80.10%
1.79090	1.0000	0.10%	80.20%	1.85190	10.0000	1.00%	81.20%
1.85720	1.0000	0.10%	81.30%	1.86980	3.0000	0.30%	81.60%
1.88280	7.0000	0.70%	82.30%	1.88630	2.0000	0.20%	82.50%
1.89540	5.0000	0.50%	83.00%	1.91760	6.0000	0.60%	83.60%
1.92410	5.0000	0.50%	84.10%	1.96310	3.0000	0.30%	84.40%
1.97370	2.0000	0.20%	84.60%	1.98880	2.0000	0.20%	84.80%
1.98960	1.0000	0.10%	84.90%	2.00670	5.0000	0.50%	85.40%
2.05580	1.0000	0.10%	85.50%	2.13800	1.0000	0.10%	85.60%
2.21350	6.0000	0.60%	86.20%	2.23060	1.0000	0.10%	86.30%
2.24090	5.0000	0.50%	86.80%	2.25180	2.0000	0.20%	87.00%
2.27270	2.0000	0.20%	87.20%	2.33410	2.0000	0.20%	87.40%
2.33920	4.0000	0.40%	87.80%	2.35510	1.0000	0.10%	87.90%
2.43060	3.0000	0.30%	88.20%	2.48160	1.0000	0.10%	88.30%
2.49340	3.0000	0.30%	88.60%	2.52830	2.0000	0.20%	88.80%
2.56410	1.0000	0.10%	88.90%	2.67860	14.0000	1.40%	90.30%
2.70680	2.0000	0.20%	90.50%	2.76680	5.0000	0.50%	91.00%
2.77780	1.0000	0.10%	91.10%	2.81990	1.0000	0.10%	91.20%
2.93220	1.0000	0.10%	91.30%	2.94120	1.0000	0.10%	91.40%
2.96850	2.0000	0.20%	91.60%	2.97270	4.0000	0.40%	92.00%
3.07020	1.0000	0.10%	92.10%	3.14360	1.0000	0.10%	92.20%
3.14690	2.0000	0.20%	92.40%	3.17460	4.0000	0.40%	92.80%
3.22250	3.0000	0.30%	93.10%	3.23180	1.0000	0.10%	93.20%
3.26090	4.0000	0.40%	93.60%	3.29860	6.0000	0.60%	94.20%
3.38100	1.0000	0.10%	94.30%	3.40240	3.0000	0.30%	94.60%
3.51560	3.0000	0.30%	94.90%	3.58580	2.0000	0.20%	95.10%
3.69320	3.0000	0.30%	95.40%	3.71520	1.0000	0.10%	95.50%
3.86470	1.0000	0.10%	95.60%	3.89610	2.0000	0.20%	95.80%
4.04630	2.0000	0.20%	96.00%	4.16670	4.0000	0.40%	96.40%
4.33880	2.0000	0.20%	96.60%	4.39560	2.0000	0.20%	96.80%
4.42740	1.0000	0.10%	96.90%	4.44080	1.0000	0.10%	97.00%
4.57520	1.0000	0.10%	97.10%	4.58840	1.0000	0.10%	97.20%
4.73760	2.0000	0.20%	97.40%	5.11360	1.0000	0.10%	97.50%
5.15870	1.0000	0.10%	97.60%	5.21030	1.0000	0.10%	97.70%
5.46880	1.0000	0.10%	97.80%	5.54030	1.0000	0.10%	97.90%
5.59010	1.0000	0.10%	98.00%	6.06060	1.0000	0.10%	98.10%
6.17330	1.0000	0.10%	98.20%	6.20300	1.0000	0.10%	98.30%
6.48150	1.0000	0.10%	98.40%	6.51150	3.0000	0.30%	98.70%
6.88410	1.0000	0.10%	98.80%	7.14290	1.0000	0.10%	98.90%
7.28740	1.0000	0.10%	99.00%	7.63890	4.0000	0.40%	99.40%
8.46560	1.0000	0.10%	99.50%	8.76620	1.0000	0.10%	99.60%
10.79550	1.0000	0.10%	99.70%	11.41300	1.0000	0.10%	99.80%
11.97920	2.0000	0.20%	100.00%	0.00000	0.0000	0.00%	100.00%

===== Simulated Chi-Square Distribution for 31 Observations =====

Number of observations: 31
Number of iterations: 1000

Chi-Square	Obs.		(1-P)	Chi-Square	Obs.		(1-P)
0.00009	4.0000	0.40%	0.40%	0.00012	1.0000	0.10%	0.50%
0.00013	3.0000	0.30%	0.80%	0.00015	2.0000	0.20%	1.00%
0.00017	2.0000	0.20%	1.20%	0.00021	9.0000	0.90%	2.10%
0.00084	1.0000	0.10%	2.20%	0.00102	1.0000	0.10%	2.30%
0.00129	2.0000	0.20%	2.50%	0.00132	2.0000	0.20%	2.70%
0.00169	6.0000	0.60%	3.30%	0.00205	8.0000	0.80%	4.10%
0.00234	7.0000	0.70%	4.80%	0.00240	7.0000	0.70%	5.50%
0.00256	2.0000	0.20%	5.70%	0.00284	10.0000	1.00%	6.70%
0.00307	2.0000	0.20%	6.90%	0.00382	8.0000	0.80%	7.70%
0.00407	5.0000	0.50%	8.20%	0.00425	2.0000	0.20%	8.40%
0.00430	2.0000	0.20%	8.60%	0.00504	1.0000	0.10%	8.70%
0.00538	1.0000	0.10%	8.80%	0.00573	1.0000	0.10%	8.90%
0.00581	10.0000	1.00%	9.90%	0.00605	9.0000	0.90%	10.80%
0.00620	11.0000	1.10%	11.90%	0.00645	3.0000	0.30%	12.20%
0.00706	16.0000	1.60%	13.80%	0.00753	10.0000	1.00%	14.80%
0.00806	4.0000	0.40%	15.20%	0.00860	10.0000	1.00%	16.20%
0.00907	10.0000	1.00%	17.20%	0.00916	1.0000	0.10%	17.30%
0.00918	5.0000	0.50%	17.80%	0.00968	5.0000	0.50%	18.30%
0.00976	1.0000	0.10%	18.40%	0.00988	2.0000	0.20%	18.60%
0.01008	5.0000	0.50%	19.10%	0.01075	1.0000	0.10%	19.20%
0.01081	1.0000	0.10%	19.30%	0.01116	2.0000	0.20%	19.50%
0.01256	6.0000	0.60%	20.10%	0.01290	1.0000	0.10%	20.20%
0.01340	7.0000	0.70%	20.90%	0.01355	14.0000	1.40%	22.30%
0.01564	3.0000	0.30%	22.60%	0.01646	6.0000	0.60%	23.20%
0.01688	3.0000	0.30%	23.50%	0.01712	4.0000	0.40%	23.90%
0.01833	5.0000	0.50%	24.40%	0.01951	4.0000	0.40%	24.80%
0.01955	5.0000	0.50%	25.30%	0.02033	1.0000	0.10%	25.40%
0.02049	3.0000	0.30%	25.70%	0.02313	3.0000	0.30%	26.00%
0.02469	3.0000	0.30%	26.30%	0.02672	6.0000	0.60%	26.90%
0.02702	6.0000	0.60%	27.50%	0.02711	2.0000	0.20%	27.70%
0.02809	1.0000	0.10%	27.80%	0.02877	1.0000	0.10%	27.90%
0.02891	1.0000	0.10%	28.00%	0.03161	1.0000	0.10%	28.10%
0.03236	7.0000	0.70%	28.80%	0.03245	4.0000	0.40%	29.20%
0.03248	2.0000	0.20%	29.40%	0.03666	2.0000	0.20%	29.60%
0.03741	3.0000	0.30%	29.90%	0.03910	1.0000	0.10%	30.00%
0.03929	7.0000	0.70%	30.70%	0.03984	4.0000	0.40%	31.10%
0.04244	1.0000	0.10%	31.20%	0.04446	5.0000	0.50%	31.70%
0.04479	2.0000	0.20%	31.90%	0.04692	1.0000	0.10%	32.00%
0.04742	3.0000	0.30%	32.30%	0.04858	3.0000	0.30%	32.60%
0.04922	4.0000	0.40%	33.00%	0.05123	1.0000	0.10%	33.10%
0.05184	1.0000	0.10%	33.20%	0.05421	8.0000	0.80%	34.00%
0.05516	3.0000	0.30%	34.30%	0.05659	2.0000	0.20%	34.50%
0.05783	8.0000	0.80%	35.30%	0.05881	1.0000	0.10%	35.40%
0.05977	4.0000	0.40%	35.80%	0.06221	3.0000	0.30%	36.10%
0.06390	1.0000	0.10%	36.20%	0.06490	1.0000	0.10%	36.30%
0.06632	5.0000	0.50%	36.80%	0.06666	3.0000	0.30%	37.10%
0.06720	1.0000	0.10%	37.20%	0.06831	8.0000	0.80%	38.00%
0.06923	1.0000	0.10%	38.10%	0.07074	2.0000	0.20%	38.30%
0.07110	13.0000	1.30%	39.60%	0.07258	5.0000	0.50%	40.10%
0.07287	9.0000	0.90%	41.00%	0.07506	10.0000	1.00%	42.00%
0.07584	1.0000	0.10%	42.10%	0.07685	10.0000	1.00%	43.10%
0.08007	5.0000	0.50%	43.60%	0.08083	2.0000	0.20%	43.80%
0.08143	1.0000	0.10%	43.90%	0.08165	1.0000	0.10%	44.00%
0.08295	8.0000	0.80%	44.80%	0.08475	1.0000	0.10%	44.90%
0.08489	4.0000	0.40%	45.30%	0.08533	1.0000	0.10%	45.40%
0.08539	2.0000	0.20%	45.60%	0.08622	1.0000	0.10%	45.70%
0.08808	1.0000	0.10%	45.80%	0.08848	10.0000	1.00%	46.80%
0.08961	4.0000	0.40%	47.20%	0.09323	2.0000	0.20%	47.40%
0.09332	4.0000	0.40%	47.80%	0.09393	1.0000	0.10%	47.90%
0.09659	3.0000	0.30%	48.20%	0.10369	3.0000	0.30%	48.50%
0.10501	6.0000	0.60%	49.10%	0.10641	6.0000	0.60%	49.70%
0.10738	1.0000	0.10%	49.80%	0.11201	9.0000	0.90%	50.70%
0.11496	1.0000	0.10%	50.80%	0.11754	3.0000	0.30%	51.10%
0.11797	1.0000	0.10%	51.20%	0.12442	1.0000	0.10%	51.30%

0.12912	1.0000	0.10%	51.40%	0.12914	5.0000	0.50%	51.90%
0.12921	4.0000	0.40%	52.30%	0.12928	1.0000	0.10%	52.40%
0.13272	3.0000	0.30%	52.70%	0.13373	1.0000	0.10%	52.80%
0.13441	4.0000	0.40%	53.20%	0.13629	2.0000	0.20%	53.40%
0.14527	2.0000	0.20%	53.60%	0.15293	5.0000	0.50%	54.10%
0.15495	4.0000	0.40%	54.50%	0.15509	1.0000	0.10%	54.60%
0.15525	4.0000	0.40%	55.00%	0.15553	1.0000	0.10%	55.10%
0.15681	8.0000	0.80%	55.90%	0.15763	1.0000	0.10%	56.00%
0.16166	2.0000	0.20%	56.20%	0.16176	1.0000	0.10%	56.30%
0.16568	3.0000	0.30%	56.60%	0.16946	2.0000	0.20%	56.80%
0.16950	2.0000	0.20%	57.00%	0.17125	8.0000	0.80%	57.80%
0.17664	2.0000	0.20%	58.00%	0.17881	5.0000	0.50%	58.50%
0.17921	5.0000	0.50%	59.00%	0.18267	8.0000	0.80%	59.80%
0.18610	4.0000	0.40%	60.20%	0.18852	2.0000	0.20%	60.40%
0.19116	4.0000	0.40%	60.80%	0.19120	1.0000	0.10%	60.90%
0.19166	5.0000	0.50%	61.40%	0.19335	5.0000	0.50%	61.90%
0.19597	1.0000	0.10%	62.00%	0.19960	1.0000	0.10%	62.10%
0.20161	1.0000	0.10%	62.20%	0.20444	4.0000	0.40%	62.60%
0.20795	6.0000	0.60%	63.20%	0.20798	1.0000	0.10%	63.30%
0.20901	1.0000	0.10%	63.40%	0.21175	4.0000	0.40%	63.80%
0.21450	2.0000	0.20%	64.00%	0.21712	3.0000	0.30%	64.30%
0.21806	1.0000	0.10%	64.40%	0.22127	1.0000	0.10%	64.50%
0.22181	6.0000	0.60%	65.10%	0.22401	2.0000	0.20%	65.30%
0.22784	7.0000	0.70%	66.00%	0.22939	1.0000	0.10%	66.10%
0.22940	1.0000	0.10%	66.20%	0.22991	1.0000	0.10%	66.30%
0.23001	3.0000	0.30%	66.60%	0.23217	1.0000	0.10%	66.70%
0.23476	1.0000	0.10%	66.80%	0.23478	3.0000	0.30%	67.10%
0.23979	1.0000	0.10%	67.20%	0.24237	1.0000	0.10%	67.30%
0.24534	4.0000	0.40%	67.70%	0.24754	2.0000	0.20%	67.90%
0.24814	6.0000	0.60%	68.50%	0.24958	2.0000	0.20%	68.70%
0.25696	1.0000	0.10%	68.80%	0.26344	1.0000	0.10%	68.90%
0.26468	2.0000	0.20%	69.10%	0.26762	1.0000	0.10%	69.20%
0.27197	1.0000	0.10%	69.30%	0.27258	2.0000	0.20%	69.50%
0.27666	3.0000	0.30%	69.80%	0.27880	1.0000	0.10%	69.90%
0.27916	6.0000	0.60%	70.50%	0.29075	1.0000	0.10%	70.60%
0.30182	7.0000	0.70%	71.30%	0.30614	1.0000	0.10%	71.40%
0.30810	1.0000	0.10%	71.50%	0.31017	1.0000	0.10%	71.60%
0.31463	2.0000	0.20%	71.80%	0.31762	4.0000	0.40%	72.20%
0.32864	3.0000	0.30%	72.50%	0.32930	1.0000	0.10%	72.60%
0.33277	3.0000	0.30%	72.90%	0.33560	1.0000	0.10%	73.00%
0.33675	1.0000	0.10%	73.10%	0.33970	1.0000	0.10%	73.20%
0.34073	1.0000	0.10%	73.30%	0.35097	1.0000	0.10%	73.40%
0.35304	1.0000	0.10%	73.50%	0.35463	3.0000	0.30%	73.80%
0.35495	2.0000	0.20%	74.00%	0.35875	2.0000	0.20%	74.20%
0.36649	5.0000	0.50%	74.70%	0.36744	2.0000	0.20%	74.90%
0.36800	1.0000	0.10%	75.00%	0.37170	1.0000	0.10%	75.10%
0.37174	1.0000	0.10%	75.20%	0.37436	3.0000	0.30%	75.50%
0.37657	1.0000	0.10%	75.60%	0.37966	4.0000	0.40%	76.00%
0.38184	1.0000	0.10%	76.10%	0.38267	5.0000	0.50%	76.60%
0.38652	1.0000	0.10%	76.70%	0.39194	5.0000	0.50%	77.20%
0.39516	2.0000	0.20%	77.40%	0.39813	1.0000	0.10%	77.50%
0.41671	1.0000	0.10%	77.60%	0.41710	1.0000	0.10%	77.70%
0.41806	6.0000	0.60%	78.30%	0.43062	2.0000	0.20%	78.50%
0.43563	8.0000	0.80%	79.30%	0.43564	4.0000	0.40%	79.70%
0.43978	3.0000	0.30%	80.00%	0.44503	2.0000	0.20%	80.20%
0.45007	3.0000	0.30%	80.50%	0.45920	1.0000	0.10%	80.60%
0.46102	2.0000	0.20%	80.80%	0.46467	5.0000	0.50%	81.30%
0.46468	2.0000	0.20%	81.50%	0.48596	1.0000	0.10%	81.60%
0.48756	1.0000	0.10%	81.70%	0.49066	1.0000	0.10%	81.80%
0.49068	1.0000	0.10%	81.90%	0.49102	4.0000	0.40%	82.30%
0.49560	2.0000	0.20%	82.50%	0.50899	4.0000	0.40%	82.90%
0.51835	1.0000	0.10%	83.00%	0.52092	1.0000	0.10%	83.10%
0.52337	1.0000	0.10%	83.20%	0.52688	2.0000	0.20%	83.40%
0.52864	2.0000	0.20%	83.60%	0.55570	3.0000	0.30%	83.90%
0.55760	1.0000	0.10%	84.00%	0.56177	1.0000	0.10%	84.10%
0.56201	5.0000	0.50%	84.60%	0.56463	7.0000	0.70%	85.30%
0.57507	2.0000	0.20%	85.50%	0.57864	1.0000	0.10%	85.60%
0.58266	3.0000	0.30%	85.90%	0.59274	2.0000	0.20%	86.10%
0.59624	2.0000	0.20%	86.30%	0.60227	4.0000	0.40%	86.70%
0.60228	1.0000	0.10%	86.80%	0.61333	1.0000	0.10%	86.90%
0.61744	1.0000	0.10%	87.00%	0.62137	1.0000	0.10%	87.10%

0.62151	1.0000	0.10%	87.20%	0.63582	3.0000	0.30%	87.50%
0.65054	1.0000	0.10%	87.60%	0.65504	2.0000	0.20%	87.80%
0.65860	3.0000	0.30%	88.10%	0.68215	1.0000	0.10%	88.20%
0.68563	1.0000	0.10%	88.30%	0.69871	1.0000	0.10%	88.40%
0.70092	1.0000	0.10%	88.50%	0.70251	2.0000	0.20%	88.70%
0.70263	5.0000	0.50%	89.20%	0.71261	1.0000	0.10%	89.30%
0.71425	1.0000	0.10%	89.40%	0.72446	1.0000	0.10%	89.50%
0.73134	5.0000	0.50%	90.00%	0.75396	1.0000	0.10%	90.10%
0.76895	1.0000	0.10%	90.20%	0.77207	5.0000	0.50%	90.70%
0.77613	1.0000	0.10%	90.80%	0.77688	2.0000	0.20%	91.00%
0.77816	1.0000	0.10%	91.10%	0.78179	3.0000	0.30%	91.40%
0.79032	2.0000	0.20%	91.60%	0.80316	1.0000	0.10%	91.70%
0.82072	2.0000	0.20%	91.90%	0.83138	1.0000	0.10%	92.00%
0.83402	1.0000	0.10%	92.10%	0.83502	1.0000	0.10%	92.20%
0.83833	1.0000	0.10%	92.30%	0.84301	2.0000	0.20%	92.50%
0.85319	2.0000	0.20%	92.70%	0.85671	2.0000	0.20%	92.90%
0.86035	2.0000	0.20%	93.10%	0.87713	1.0000	0.10%	93.20%
0.89215	1.0000	0.10%	93.30%	0.90504	1.0000	0.10%	93.40%
0.91041	1.0000	0.10%	93.50%	0.93561	3.0000	0.30%	93.80%
0.96538	2.0000	0.20%	94.00%	0.97110	1.0000	0.10%	94.10%
1.00893	1.0000	0.10%	94.20%	1.01202	2.0000	0.20%	94.40%
1.01797	1.0000	0.10%	94.50%	1.06509	1.0000	0.10%	94.60%
1.06891	1.0000	0.10%	94.70%	1.08007	1.0000	0.10%	94.80%
1.09429	1.0000	0.10%	94.90%	1.10250	1.0000	0.10%	95.00%
1.11514	1.0000	0.10%	95.10%	1.12790	1.0000	0.10%	95.20%
1.13380	1.0000	0.10%	95.30%	1.13610	4.0000	0.40%	95.70%
1.18596	1.0000	0.10%	95.80%	1.19126	1.0000	0.10%	95.90%
1.19377	1.0000	0.10%	96.00%	1.19746	1.0000	0.10%	96.10%
1.20443	1.0000	0.10%	96.20%	1.23528	1.0000	0.10%	96.30%
1.26265	1.0000	0.10%	96.40%	1.27167	3.0000	0.30%	96.70%
1.28541	1.0000	0.10%	96.80%	1.36260	1.0000	0.10%	96.90%
1.37110	2.0000	0.20%	97.10%	1.46251	2.0000	0.20%	97.30%
1.53610	1.0000	0.10%	97.40%	1.61352	1.0000	0.10%	97.50%
1.62346	1.0000	0.10%	97.60%	1.64667	1.0000	0.10%	97.70%
1.66767	1.0000	0.10%	97.80%	1.71902	1.0000	0.10%	97.90%
1.73169	1.0000	0.10%	98.00%	1.74594	1.0000	0.10%	98.10%
1.75021	1.0000	0.10%	98.20%	1.77672	1.0000	0.10%	98.30%
1.79619	1.0000	0.10%	98.40%	1.85086	1.0000	0.10%	98.50%
1.93629	1.0000	0.10%	98.60%	1.94829	1.0000	0.10%	98.70%
1.97134	1.0000	0.10%	98.80%	2.07410	1.0000	0.10%	98.90%
2.10277	1.0000	0.10%	99.00%	2.13891	1.0000	0.10%	99.10%
2.15745	1.0000	0.10%	99.20%	2.18776	1.0000	0.10%	99.30%
2.38196	2.0000	0.20%	99.50%	2.52930	1.0000	0.10%	99.60%
2.82969	1.0000	0.10%	99.70%	2.86044	1.0000	0.10%	99.80%
3.12841	1.0000	0.10%	99.90%	3.27294	1.0000	0.10%	100.00%