Data Security in European Healthcare Information Systems

by

Steven Marcus Furnell
B.Sc. (Hons)

A thesis submitted to the University of Plymouth in partial fulfilment for the degree of

DOCTOR OF PHILOSOPHY

School of Electronic, Communication and Electrical Engineering
Faculty of Technology

In collaboration with
Plymouth Health Authority

June 1995
Data Security in European Healthcare Information Systems

Steven Marcus Furnell
B.Sc. (Hons)

This thesis considers the current requirements for data security in European healthcare systems and establishments. Information technology is being increasingly used in all areas of healthcare operation, from administration to direct care delivery, with a resulting dependence upon it by healthcare staff. Systems routinely store and communicate a wide variety of potentially sensitive data, much of which may also be critical to patient safety. There is consequently a significant requirement for protection in many cases.

The thesis presents an assessment of healthcare security requirements at the European level, with a critical examination of how the issue has been addressed to date in operational systems. It is recognised that many systems were originally implemented without security needs being properly addressed, with a consequence that protection is often weak and inconsistent between establishments. The overall aim of the research has been to determine appropriate means by which security may be added or enhanced in these cases.

The realisation of this objective has included the development of a common baseline standard for security in healthcare systems and environments. The underlying guidelines in this approach cover all of the principal protection issues, from physical and environmental measures to logical system access controls. Further to this, the work has encompassed the development of a new protection methodology by which establishments may determine their additional security requirements (by classifying aspects of their systems, environments and data). Both the guidelines and the methodology represent work submitted to the Commission of European Communities SEISMED (Secure Environment for Information Systems in MEDicine) project, with which the research programme was closely linked.

The thesis also establishes that healthcare systems can present significant targets for both internal and external abuse, highlighting a requirement for improved logical controls. However, it is also shown that the issues of easy integration and convenience are of paramount importance if security is to be accepted and viable in practice. Unfortunately, many traditional methods do not offer these advantages, necessitating the need for a different approach.

To this end, the conceptual design for a new intrusion monitoring system was developed, combining the key aspects of authentication and auditing into an advanced framework for real-time user supervision. A principal feature of the approach is the use of behaviour profiles, against which user activities may be continuously compared to determine potential system intrusions and anomalous events.

The effectiveness of real-time monitoring was evaluated in an experimental study of keystroke analysis - a behavioural biometric technique that allows an assessment of user identity from their typing style. This technique was found to have significant potential for discriminating between impostors and legitimate users and was subsequently incorporated into a fully functional security system, which demonstrated further aspects of the conceptual design and showed how transparent supervision could be realised in practice.

The thesis also examines how the intrusion monitoring concept may be integrated into a wider security architecture, allowing more comprehensive protection within both the local healthcare establishment and between remote domains.
## CONTENTS

Abstract i  
Contents ii  
List of Figures xi  
List of Tables xiv  
Acknowledgements xvi  
Declaration xviii  
Glossary of Abbreviations xix  
Glossary of Terms xxi  

### Chapter 1: Introduction and Overview 1

1.1 Introduction 2  
1.2 Aims and objectives of the research 4  
1.3 Thesis Structure 7  

### Chapter 2: The need for Security in Healthcare 13

2.1 An overview of Information Security 14  
2.2 Current trends in European healthcare and informatics 17  
  2.2.1 Healthcare in the European Union 17  
  2.2.2 Use of Information Technology 19  
2.3 Security requirements in healthcare 21  
2.4 Special factors of the healthcare environment 24  
  2.4.1 Confidentiality 25  
  2.4.2 Integrity 27  
  2.4.3 Availability 29
5.1.1 Traditional risk analysis
5.1.2 Requirements for a profiling methodology

5.2 Elements of the methodology
5.2.1 Computer Configurations
5.2.2 Operational Environments
5.2.3 Data Sensitivity
  5.2.3.1 Overview of healthcare data requirements
  5.2.3.2 A Healthcare Generic Data Model
  5.2.3.3 Data Use
  5.2.3.4 Approach to sensitivity rating

5.3 Countermeasure selection
5.4 Formal stages of the methodology
  5.4.1 Determining Data Sensitivity
5.5 An example of methodology implementation
5.6 Extension of the methodology
  5.6.1 Enhanced system classification
  5.6.2 Expert system
5.7 Conclusions

Chapter 6: Improving system security in healthcare
6.1 Introduction
6.2 Intrusions in healthcare systems
6.3 An overview of Intrusions and Intruders
6.4 Traditional approaches to Intrusion Detection
6.5 Advanced approaches to Intrusion Detection
  6.5.1 System auditing
Chapter 6: Behavioural Observations

6.5.2 Initial user identification and authentication

6.5.3 User behaviour profiling
  6.5.3.1 User class profiling
  6.5.3.2 User-specific profiling

6.5.4 Generic intrusion indicators and rules

6.6 Generic behaviour profiles for healthcare users
  6.6.1 Introduction
  6.6.2 Categorisation of HCE staff
  6.6.3 Questionnaire content
  6.6.4 Questionnaire distribution
  6.6.5 Results and general observations
  6.6.6 Limitations
  6.6.7 Conclusions

6.7 A survey of existing Intrusion Detection Systems

6.8 Conclusions

Chapter 7: Conceptual design of an Intrusion Monitoring System

7.1 Introduction

7.2 Operating Cycle

7.3 Response to suspected intrusions

7.4 IMS Architecture
  7.4.1 Anomaly Detector
    7.4.1.1 Maintenance of an intrusion alert status
    7.4.1.2 Restriction of user activities
    7.4.1.3 Suspension of supervision
  7.4.2 Profile Refiner
7.4.3 Recorder 199
7.4.4 Archiver 200
7.4.5 Collector 201
7.4.6 Responder 202
7.4.7 Communicator 203
7.4.8 Controller 203
7.4.9 Profiles 204

7.5 IMS Implementation 206
7.5.1 PC implementation 206
7.5.2 Minicomputer implementation 207

7.6 Advantages and disadvantages of the IMS approach 209
7.6.1 Advantages 209
7.6.2 Disadvantages 211

7.7 Intrusion monitoring in a wider security architecture 213
7.7.1 The Comprehensive Integrated Security System 213
7.7.2 Intrusion monitoring in the CISS framework 216
7.7.3 Distribution of IMS functionality within the CISS 217

7.8 Conclusions 223

Chapter 8: Real-time supervision using Keystroke Analysis 225
8.1 Introduction 226
8.2 Keystroke Analysis Concepts 228
8.2.1 Typing characteristics 229
8.2.2 Categories of typist 230
8.2.3 Authentication strategies 231
8.2.3.1 Static Identifier 231
8.2.3.2 Dynamic Identifier

8.2.4 Evaluation of effectiveness

8.3 Implementation considerations for a Keystroke Analysis System

8.3.1 Typing characteristics evaluated

8.3.2 Keystrokes selected for analysis

8.3.3 Creation of Keystroke Profiles

8.3.3.1 Selection of a Reference Text

8.3.3.2 Filtering of profile sample timings

8.3.4 Authentication assessment

8.4 The experimental Keystroke Analysis System

8.4.1 Implementation environment

8.4.2 System Modules

8.4.2.1 Profiler

8.4.2.2 Sampler

8.4.2.3 Monitor

8.5 Full Keystroke Analysis study

8.5.1 Test subjects

8.5.2 Experimental procedure

8.5.3 Results and analysis

8.6 Potential Enhancements

8.6.1 Impostor identity suggestion

8.6.2 Increased profile specificity

8.6.3 Detection of subject impairment

8.6.4 Neural Network implementation

8.7 Potential Problems

8.7.1 Consistency of users
8.7.2 Mimicry 276
8.7.3 User acceptance 277
8.7.4 Accuracy of keystroke timings 278
8.7.5 General applicability 280

8.8 Conclusions 280

Chapter 9: An IMS Demonstrator System 283

9.1 Enhancement of the Keystroke Analyser 284

9.2 IMS Client Implementation 287
   9.2.1 System Configuration Auditing 287
   9.2.2 Virus Scanning 288
   9.2.3 User Identification and initial Authentication 291
   9.2.4 Implementation and operation of background supervision 292
   9.2.5 Challenges and Session Supervision 295
   9.2.6 User logout / Session termination 296

9.3 IMS Host implementation 298
   9.3.1 Menu Options & General Functionality 298
      9.3.1.1 Monitoring Options 298
      9.3.1.2 Profiling Options 299
      9.3.1.3 System Management 300
   9.3.2 Intrusion Monitoring and Detection 301
   9.3.3 Event Auditing 304
   9.3.4 Internal Communications 305
   9.3.5 Host - Client Communication 306
   9.3.6 Profiling Sub-system 307

9.4 Implementation constraints and potential enhancements 309
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 2.1</td>
<td>Perspectives on information system protection</td>
<td>16</td>
</tr>
<tr>
<td>Fig. 2.2</td>
<td>SEISMED Reference Centres, partners and collaborators</td>
<td>39</td>
</tr>
<tr>
<td>Fig. 3.1</td>
<td>Main applications</td>
<td>47</td>
</tr>
<tr>
<td>Fig. 3.2</td>
<td>General availability expectations</td>
<td>48</td>
</tr>
<tr>
<td>Fig. 3.3</td>
<td>Relative adoption of security techniques</td>
<td>49</td>
</tr>
<tr>
<td>Fig. 3.4</td>
<td>Methods of authentication</td>
<td>50</td>
</tr>
<tr>
<td>Fig. 3.5</td>
<td>Security Incidents in European HCEs</td>
<td>51</td>
</tr>
<tr>
<td>Fig. 4.1</td>
<td>Approaches to existing systems security</td>
<td>54</td>
</tr>
<tr>
<td>Fig. 4.2</td>
<td>HCE target audiences</td>
<td>64</td>
</tr>
<tr>
<td>Fig. 5.1</td>
<td>Elements of an information system</td>
<td>75</td>
</tr>
<tr>
<td>Fig. 5.2</td>
<td>Existing Systems Protection Methodology Overview</td>
<td>76</td>
</tr>
<tr>
<td>Fig. 5.3</td>
<td>Computer Configuration groups</td>
<td>77</td>
</tr>
<tr>
<td>Fig. 5.4</td>
<td>Factors of data sensitivity</td>
<td>80</td>
</tr>
<tr>
<td>Fig. 5.5</td>
<td>General care activity flow</td>
<td>81</td>
</tr>
<tr>
<td>Fig. 5.6</td>
<td>Classifications of medical data</td>
<td>82</td>
</tr>
<tr>
<td>Fig. 5.7</td>
<td>Healthcare Generic Data Model</td>
<td>85</td>
</tr>
<tr>
<td>Fig. 5.8</td>
<td>Patient Administration System mapping</td>
<td>96</td>
</tr>
<tr>
<td>Fig. 5.9</td>
<td>Countermeasure selection summary</td>
<td>104</td>
</tr>
<tr>
<td>Fig. 5.10</td>
<td>Data Sensitivity Assessment</td>
<td>107</td>
</tr>
<tr>
<td>Fig. 5.11</td>
<td>Expert System Structure and Interaction</td>
<td>117</td>
</tr>
</tbody>
</table>
Fig. 8.11: False acceptance and successful impersonation of each subject

Fig. 8.12: Keystrokes before impostor detection

Fig. 9.1: Configuration of IMS demonstrator system

Fig. 9.2: Response to challenge signal

Fig. 9.3: Demonstrator menu structure

Fig. 9.4: IMS Demonstrator, Keystroke Monitoring Interface

Fig. 9.5: IMS Host file usage

Fig. 9.6: IMS Host - Client Communication

Fig. 9.7: IMS Profiling Sub-System

Fig. 9.8: Polling of active IMS Clients in network monitoring system

Fig. 10.1: Symmetric Encryption

Fig. 10.2: Asymmetric Encryption

Fig. 10.3: The Naming Authority hierarchy

Fig. 10.4: Logical certification hierarchy

Fig. 10.5: TTP infrastructure at the international and national levels

Fig. 10.6: TTP infrastructure at the national and local levels

Fig. 10.7: Potential inter-domain HCE communications

Fig. 10.8: Secure inter-domain communication
LIST OF TABLES

Table 4.1: Principles of Existing Systems Security 58

Table 5.1: Computer Configuration countermeasure groups 78
Table 5.2: Operational Environment categorisation 79
Table 5.3: Operational Environment countermeasure groups 79
Table 5.4: General categories of medical data usage 98
Table 5.5: Sensitivity ratings for data disclosure 101
Table 5.6: Sensitivity ratings for data denial, modification & destruction 101
Table 5.7: Derivation of sensitivity ratings 101
Table 5.8: Formal stages of the protection methodology 105
Table 5.9: Stages of data sensitivity assessment 106

Table 6.1: Categories of system abuser 130
Table 6.2: Auditable characteristics of PC configuration 141
Table 6.3: Potential IMS profile characteristics 149
Table 6.4: Scope of intrusion detection methods 154
Table 6.5: Examples of generic intrusion indicators 156
Table 6.6: Rationale of Survey Questions 165
Table 6.7: Responses to profiling questionnaire 168
Table 6.8: Healthcare “class” profiles 169
Table 6.9: Overview of existing intrusion detection systems 175

Table 7.1: Alert status threshold table 195
Table 7.2: Relationship of IMS modules to CISS agents 218
Table 7.3: IMS data storage in the SMIB 219
Table 8.1: Typist Skill Categorisation 231
Table 8.2: Effect of profile sample size 244
Table 8.3: Pseudo-code for replacement interrupt routines 254
Table 8.4: Keystroke analyser files 259
Table 8.5: Profiled Performance of Test Subjects 261
Table 8.6: Overview of impostor detection for test sample 1 265
Table 8.7: Overview of impostor detection for test sample 2 266
Table 8.8: Figures for subject false acceptance and successful impersonation 267
ACKNOWLEDGEMENTS

The work presented in this thesis represents the results of a three year investigation into various aspects of information systems security in the healthcare field. All work was undertaken within the Network Research Group in the School of Electronic, Communication and Electrical Engineering, University of Plymouth, Plymouth, England.

I would principally like to acknowledge the contributions of the following people:

- Peter Sanders, my Director of Studies, whose help, knowledge and support throughout the project provided direction for the work and ensured that the best opportunities were always available;

- Dr Colin Stockel, my Supervisor, who provided valuable advice and guidance throughout the project and was an unflinching source of ideas on virtually all matters (from subjects as diverse as the research programme itself to the assassination of President John F. Kennedy!);

- Dr Nick Gaunt, of Plymouth Health Authority and the Department of Healthcare Informatics at Derriford Hospital, who provided further valuable input to the research programme and significant assistance in respect to the SEISMED work.

I would also like to acknowledge the assistance of various members of the SEISMED consortium who contributed ideas and opinions relating to the security guidelines, the generic protection methodology and the initial findings from the keystroke analysis study.
Thanks are also due to the various friends and colleagues who spared their time to participate as test subjects in the keystroke analysis experiments, along with special thanks to Dr Simon Shepherd of the University of Bradford for his valuable assistance during the initial development of the keystroke analyser.

Finally, thanks must go to my friends and family for their support and encouragement over the course of the last three years.
DECLARATION

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award.

This study was financed with funding from the Commission of European Communities SEISMED project and was carried out in collaboration with Plymouth Health Authority.

Relevant conferences and SEISMED project meetings were regularly attended (at which work was frequently presented) and a number of external establishments were visited for consultation purposes. In addition, several papers were prepared for publication, details of which are listed in the appendices.

Signed ..........................................................

Date 3/8/95
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIM</td>
<td>Advanced Informatics in Medicine</td>
</tr>
<tr>
<td>CA</td>
<td>Certification Authority</td>
</tr>
<tr>
<td>CCTA</td>
<td>Central Computer and Telecommunications Agency</td>
</tr>
<tr>
<td>CEC</td>
<td>Commission of the European Communities</td>
</tr>
<tr>
<td>CISS</td>
<td>Comprehensive Integrated Security System</td>
</tr>
<tr>
<td>CRAMM</td>
<td>CCTA Risk Analysis and Management Methodology</td>
</tr>
<tr>
<td>DES</td>
<td>Data Encryption Standard</td>
</tr>
<tr>
<td>DOS</td>
<td>Disk Operating System (abbreviation of MS-DOS)</td>
</tr>
<tr>
<td>DTI</td>
<td>Department of Trade and Industry</td>
</tr>
<tr>
<td>EHCR</td>
<td>Electronic Healthcare Record</td>
</tr>
<tr>
<td>EU</td>
<td>European Union</td>
</tr>
<tr>
<td>FAR</td>
<td>False Acceptance Rate</td>
</tr>
<tr>
<td>FRR</td>
<td>False Rejection Rate</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>HCE</td>
<td>Healthcare Establishment</td>
</tr>
<tr>
<td>HCP</td>
<td>Healthcare Professional</td>
</tr>
<tr>
<td>IMS</td>
<td>Intrusion Monitoring System</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organisation</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>MA</td>
<td>Monitoring Agent (part of the CISS architecture)</td>
</tr>
<tr>
<td>MAC</td>
<td>Message Authentication Code</td>
</tr>
<tr>
<td>MS-DOS</td>
<td>Microsoft Disk Operating System</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>NA</td>
<td>Naming Authority</td>
</tr>
<tr>
<td>NHS</td>
<td>National Health Service</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>OSI</td>
<td>Open Systems Interconnection</td>
</tr>
<tr>
<td>PAM</td>
<td>Professions Allied to Medicine</td>
</tr>
<tr>
<td>PAS</td>
<td>Patient Administration System</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer (IBM compatible)</td>
</tr>
<tr>
<td>RSA</td>
<td>Rivest, Shamir and Adleman (method of asymmetric encryption)</td>
</tr>
<tr>
<td>SEISMED</td>
<td>Secure Environment for Information Systems in Medicine</td>
</tr>
<tr>
<td>SMC</td>
<td>Security Management Centre</td>
</tr>
<tr>
<td>SMIB</td>
<td>Security Management Information Base (part of the CISS architecture)</td>
</tr>
<tr>
<td>THIS</td>
<td>Trusted Health Information Systems (INFOSEC project)</td>
</tr>
<tr>
<td>TSR</td>
<td>Terminate, Stay Resident (a class of PC program)</td>
</tr>
<tr>
<td>TTP</td>
<td>Trusted Third Party</td>
</tr>
<tr>
<td>WAN</td>
<td>Wide Area Network</td>
</tr>
</tbody>
</table>
The following pages present a series of definitions for the key terminology used within the thesis.

**Access Control**

The prevention of unauthorised use of a resource, including the prevention of use of a resource in an unauthorised manner.

**Alert Status**

A rating of the level of *anomaly* in a user session / process that is maintained by the Intrusion Monitoring System (IMS).

**Anomaly**

A suspicious or unauthorised system activity.

**Audit Trail**

The historic data and information which are available for examination in order to prove the correctness and *integrity* with which the agreed security procedures related to a key or transaction(s) have been followed and which allows breaches in *security* to be detectable.

**Authentication**

The verification of a claimed identity.
Availability

The property of data and systems being accessible and usable upon demand by an authorised entity.

Baseline

The minimum acceptable level of security necessary to protect a system.

Certificate

The document that binds an entity’s unique name and its public key, along with some other information, rendered unforgable by the digital signature of the certification authority that issued it.

Certification Authority

An authority trusted by one or more users to create and sign certificates.

Checksum

A value calculated from items of data that may be used to verify that the data has not been altered.

Clandestine user

A class of system intruder, referring to a user who evades access controls and auditing.

Computer Misuse

Unauthorised or improper use of information systems or IT facilities, including the abuse of privileges by authorised users.
Confidentiality

The property that information is not made available or disclosed to unauthorised individuals, entities or processes.

Countermeasure

A mechanism or procedure placed in a system environment to reduce one or more elements of risk (i.e. threats, impact or vulnerability).

Digital Signature

Data appended to, or a cryptographic transformation of, a data unit that allows the recipient to prove the source and integrity of the data and protecting against forgery.

Encryption

A process of disguising information so that it cannot be understood by an unauthorised person.

Existing System

A system that is already operational within a Healthcare Establishment.

External Penetrator

A class of system intruder, referring to an outsider who attempts or gains unauthorised access to the system.

False Acceptance Rate (FAR)

The proportion of cases in which impostors are falsely authenticated by the system (also referred to as Impostor Pass Rate).
**False Rejection Rate (FRR)**

The proportion of cases in which legitimate users are rejected by the system (also referred to as False Alarm Rate).

**Healthcare Establishment (HCE)**

An establishment where medical services are rendered or health education, healthcare research or medical training or prevention activities are conducted.

**Health Information System**

A system that processes health data.

**Impact**

The effect of a failure to preserve confidentiality, integrity and/or availability. Impact may be one of four types: Disclosure, Modification, Destruction or Denial.

**Information System**

A collection of people, procedures and equipment maintained to gather, record, process, store, retrieve and display information.

**Integrity**

The property that systems or data have not been altered or destroyed in an unauthorised manner.
**Internal Penetrator**

A class of system intruder, referring to authorised users of the system who access data, resources or programs to which they are not entitled. Sub-categorised into *masqueraders* and *clandestine users*.

**Intruder**

An entity (e.g. a user or process) performing anomalous activities. Intruders may be subclassified into *external penetrators, internal penetrators, misfeasors* and *malicious processes*.

**Intrusion**

The situation when a series of *anomalies* are judged to represent a genuine attempt to compromise the system.

**Logical Security**

System-based protection mechanisms that allow system managers to control access to and use of HCE applications and data (and thereby help preserve their *confidentiality, integrity* and *availability*).

**Malicious Process**

A class of computer program, such as a *virus*, *worm* or *Trojan Horse*, that can intentionally damage or disrupt the operation of an computer system.

**Masquerader**

A class of system intruder, referring to a user who operates under the identity of another user.
Mean

The arithmetic average of a set of values.

Misfeasor

A class of system intruder, referring to a user who is authorised to use the system and resources accessed, but misuses these privileges.

Non-repudiation

A security service providing a message recipient with a proof that the claimed origin of the message is genuine.

Operating System

The computer program that performs basic housekeeping functions of the system (e.g. maintaining lists of files, running programs). Common operating systems include MS-DOS, MS Windows, VMS and Unix.

Password

Confidential authentication information, usually composed of a string of characters.

Personnel Security

The procedures established to ensure that all personnel who have access to any sensitive information have the required authorities as well as all appropriate clearances.
**Physical Security**

The measures used to provide physical protection of resources against deliberate and accidental threats.

**Primary Care**

The initial source of healthcare at the community level, conducted by a general practitioner and his/her team.

**Privacy**

The right of individuals to control or influence what information related to them may be collected and stored and by whom that information may be disclosed.

**Private Key**

In a public key cryptosystem, the key that is known only to the legitimate user and used for decryption or signature generation.

**Profile**

A description of user behaviour (in terms of the way that they typically use computer systems) that can be used for monitoring and supervision purposes. Profiles may be specified at a generic (class) level or on an individual basis and may contain information on one or more characteristics of system usage.

**Public Key**

In a public key cryptosystem, the key that is made publicly available and used for encryption or signature verification.
Reference Text

The text passage used as the basis for the creation of user typing profiles in the keystroke analyser system.

Risk

This is a method of determining the threats and vulnerability of a particular asset.

Risk Analysis

Involves the identification and assessment of risk against assets.

Scancode

An identification number associated with each key on a PC keyboard. Each key has two associated codes; one to denote key depression (the “make” code) and another to denote release (the “break” code).

Sensitive Data

This refers to data for which unauthorised disclosure, modification or unavailability could adversely affect patients or the healthcare establishment.

Security

The combination of confidentiality, integrity and availability.

Smart Card

A machine-readable card, normally containing a microprocessor, which is capable of holding data and performing computations.
**Standard Deviation**

A statistic used as a measure of dispersion in a distribution. The square root of the arithmetic average of the squares of the deviations from the *mean*.

**System Security Policy**

The set of laws, rules and practices that regulate how sensitive information and other resources are managed, protected and distributed within a specific system.

**Terminate Stay Resident (TSR)**

A special class of PC program in which some of the code remains resident in memory after program termination and can subsequently be activated by appropriate triggers. This feature may be used to provide a rudimentary background processing facility and is the basis for the transparent operation of the IMS Demonstrator.

**Threat**

A potential violation of *security*.

**Trojan Horse**

An executable program that claims and / or appears to perform some useful or harmless function, but also conceals a malicious purpose.

**Trusted Third Party (TTP)**

A person or organisation entrusted by a domain of users to provide a security service and who is independent of the communicating parties.
**Virus**

A class of malicious software program that has the ability to self-replicate and "infect" parts of the *operating system* or application programs, with the potential to cause loss of or damage to data.

**Vulnerability**

The likeliness of a threat to become reality.

**Worm**

A self-replicating program designed to "breed" within computer systems, but lacking the potentially destructive "payload" element of a *virus*. 
CHAPTER 1

Introduction and Overview
1.1 Introduction

Over the last three decades Information Technology (IT) has become progressively more widespread in all areas of society. As a consequence, there are now very few people who do not encounter this technology, in some form, as part of their everyday lives. Future improvements in computer processing power and communication networks should ensure that this trend can continue.

The advancement of IT has been accompanied by a progressive improvement in the reliability of systems. Whereas non-operational systems were an accepted and rather routine occurrence in earlier generations, such problems are now much less frequent. The net effect of this is that confidence and trust in IT has grown, leading to a generally uncritical reliance upon computers in most sections of society. As a result, computer systems have been entrusted to handle increasingly more important functions and information (often without any further checks so long as operations and data appear reasonable). This trend again appears set to continue as the further advances open up yet more opportunities.

The combination of these points serves to make the protection of systems a vital concern, and necessitates that security is now considered an essential aspect of the information technology field. The introduction of security seeks to eliminate or, more realistically, reduce the vulnerability to any risks that may be present. At a general level, protection must encompass the computer system and everything associated with it (e.g. from the computer unit itself to the building in which it is housed). Most important, however, is normally the protection of the data stored by the systems.
Information systems are now commonplace in the healthcare field, with computers being routinely used in all levels of healthcare establishment (HCE), from primary care up to general hospitals. An increasing amount of information is handled primarily (and sometimes exclusively) by computerised systems. The nature of this information is very diverse, with much being considered sensitive and, in some cases, critical to human life. In addition, the more commercial environment of healthcare (particularly in the UK) means that the protection of financial data is also becoming very important. As such, the effects of loss, damage or disclosure of information are not finite and could have various wide ranging consequences for both the HCE and its patients. There is consequently an increasing concern within the healthcare communities of Europe for the security of information that is stored and transmitted within their computer systems.

Security will be required to some extent in all types of HCE (note that for the purpose of this study a HCE is considered to be any establishment where medical services are offered or where training, research or prevention activities are conducted (Katsikas and Gritzalis 1994)). However, healthcare systems and environments are generally very large and complex, with many possible points of access and different operating procedures. These factors pose problems when trying to ensure that data is properly protected.

Many systems do exist in which some attempt has been made to address security, but these have met with varying degrees of success and a general observation is that the most effective cases are those where security needs were recognised from the outset. However, many HCEs have a significant investment in operational systems where security has not
been properly addressed, even though in other respects the features and performance of the systems may totally satisfy user requirements. It is obviously important that security requirements in such systems are not overlooked and a key issue is, therefore, how protection can be added or enhanced in these cases.

A broad overview of information security and a more detailed examination of the specific requirements in the healthcare sector is presented in chapter 2.

1.2 Aims and objectives of the research

This study is concerned with the issue of data security in modern healthcare establishments and information systems. It recognises the importance of security in providing a foundation for the future integration and harmonisation of the healthcare community at a European level.

The field of healthcare security encompasses a huge range of issues and it is consequently possible to identify a significant number of areas worthy of investigation. It was, therefore, necessary to determine an appropriate boundary for the research and, given the observations in the previous section, this study specifically addressed the investigation and development of suitable security methods for operational healthcare systems.

For the purposes of this discussion, the overall research programme can be divided into two key phases. The first of these was concerned with the production of viable recommendations for the standardisation and improvement of HCE security at a general level, based upon existing techniques and technologies. The second phase was related to a
more specific practical investigation, with the aim of evaluating potential methods for enhancing protection technologies within existing systems.

A principal objective of the first phase was to develop a suitable generic model for data storage and flows within the European Health Communities that would allow the necessary security requirements to be indicated for their existing systems. This was then extended to include the development of a full protection methodology, incorporating not only a data model but also a means of classifying other system components.

With regard to the requirement to address practical implementation, the key issues were seen as being convenience and easy integration (requiring a security system that could be implemented as an overlay service to existing systems whilst still providing a high degree of transparency to the end-user). In this way, the provision of security should not necessitate major changes to existing applications or the way in which they are used. As such, the chosen approach was an investigation of real-time supervision using a technique considered to be suitable for use in HCEs. However, this stage of the work was viewed as being less specifically tied to the healthcare environment, as it was considered that the practical techniques under investigation would almost certainly be applicable in many other sectors as well. As such, the discussion at this level does not limit itself to the healthcare domain.

The full objectives of the research programme can be more formally listed as follows:

1. to assess the general need for information systems security within healthcare establishments;
2. to assess the current use of security in European healthcare systems and identify weaknesses and requirements;

3. to determine practical and viable methods of enhancing security in existing healthcare systems;

4. to develop a mechanism by which HCEs may determine their own security requirements, considering the sensitivity of existing systems in terms of both system and data elements;

5. to suggest means of enhancing the protection technologies used in healthcare systems and evaluate the effectiveness of the selected approach(es);

6. to show how the chosen technique(s) could be implemented in practice using a demonstrator system;

7. to examine how the suggested techniques could be incorporated into a wider and more comprehensive security system, which would also account for future trends in healthcare information system usage, such as increasing inter-establishment data exchange.

These objectives correspond to the general sequence of the material presented in the subsequent chapters of the thesis, as will be discussed in the next section.
The research has involved significant liaison with healthcare professionals (particularly during the early stages of the work). The majority of this consultation occurred within the context of the AIM SEISMED project (as will be described in chapter 2), which was concerned with the improvement of healthcare security at a European level.

1.3 Thesis Structure

This thesis describes the research leading to the formulation of a suitable security strategy for addressing some of the requirements of existing healthcare systems within Europe. The investigation began at a general level, with an approach that encompassed all of the main security considerations applicable in healthcare, before proceeding to identify a particular class of technical approach and describing the conduct of a practical evaluation.

Chapter 2 begins by presenting a general overview of the key issues associated with information security. The chapter also highlights the increasing use and potential dependence upon information systems within the healthcare community, and the consequent demands for security which this brings. This is then followed by an examination of the specific requirements that exist in the healthcare field and a discussion of why healthcare is somewhat different from many other environments where security is required. This chapter also introduces the AIM SEISMED project and explains the close links to the initial phases of the research.

Chapter 3 then considers the results of a European survey that examined the current security practice and attitudes within the medical community. This serves to underline some
of the points made in the previous chapter regarding the need for security in existing systems. An analysis is presented which highlights the weaknesses and inconsistencies in the current scenario, pointing to the requirement for a consistent European approach.

Chapter 4 begins to address the issue of how such a consistent approach may be realised by introducing the concept of a baseline level for healthcare security. This highlights the need for a standardised approach in the protection of existing systems. A number of general recommendations are presented, based upon a new set of information security guidelines that have been developed specifically for the medical community. The development of these guidelines was an integral part of the first phase of the research and this chapter describes their general purpose, the target audiences and the main points from a series of protection principles.

Chapter 5 then proceeds to present the main product of the first phase of the research, namely a generic protection methodology specifically tailored to the needs of the healthcare environment. It explains the need for a simplified system of security profiling which can be applied by existing staff as an alternative to extensive risk analysis investigations. This leads into a detailed description of the methodology that has been developed, with a general overview of the key elements and descriptions of the formal stages. The methodology is based upon the classification of existing information systems (i.e. identifying aspects of the computers, environments and data involved) and subsequent selection of appropriate security countermeasures. A worked example is used to illustrate how the approach would be applied in a typical healthcare information system scenario. A significant section of the chapter is devoted to the description of a healthcare data model that was developed as part
of the methodology. This attempts to encompass all of the principal data requirements in European HCEs and thereby allows a relatively simple means of assessing the data sensitivity of healthcare information systems.

With the guidelines and methodology having been based on existing protection techniques, chapter 6 provides the link to the more practical elements of the research by identifying a requirement to enhance the protection technology itself. To this end, an overview of real-time supervision and intrusion monitoring is presented, along with an explanation of how it improves upon conventional user authentication and audit trails in healthcare systems. The different types of potential system intrusion are categorised, with brief descriptions in each case. Supporting evidence is also given to show cases where such incidents have occurred in the healthcare field, providing further justification for such a system. Various approaches to intrusion detection are then considered, including the development of user behaviour profiles, monitoring for generic intrusion indicators and auditing of system-related factors (e.g. changes to the hardware and software configuration). A detailed examination of potential techniques is given in each case. This includes a specific link to the healthcare field in the form of a series of generic behaviour profiles for different classes of healthcare user. These were developed from the results of a survey of healthcare personnel conducted within a local HCE. Finally, a brief summary of previous work in the intrusion detection field (based upon systems operating in non-healthcare domains) is also presented.

Having established the main concepts and fundamental elements of system activity that can be monitored, chapter 7 then proceeds to present the comprehensive conceptual design for an intrusion monitoring system (IMS), with appropriate links to show how it may be applied
Chapter 1: Introduction and Overview

in the healthcare environment. The main discussion concentrates upon how intrusion detection could realistically be achieved, with a detailed description of a proposed system architecture (based upon the idea of a central monitoring host, with a series of local clients collecting system activity data and responding to detected anomalies), explaining how the different components would be linked and function together. The chapter also explains how intrusion monitoring could form part of a more general security architecture for local domains. This makes reference to the concept of a Comprehensive Integrated Security System (CISS), which has been defined in a previous research programme (Shepherd 1992) and shows how the intrusion monitoring system would represent a key element of a local Security Management Centre (SMC).

Chapter 8 contains a comprehensive investigation of one of the potential intrusion monitoring techniques identified in chapter 6, namely user identity verification using keystroke analysis. It begins with a detailed examination of the concept and the various factors that need to be considered for successful implementation (e.g. distinctive characteristics of user typing, creation of representative profiles and strategies for implementation). The chapter subsequently proceeds to present the results from an experimental study that has been conducted, allowing an assessment of the suitability of the technique for healthcare implementation.

Chapter 9 describes a further stage of practical development that was undertaken, whereby the keystroke analysis technique was incorporated into the more comprehensive supervision framework described in chapter 7. This led to the creation of an IMS demonstrator system which succeeded in fulfilling a subset of the key elements from the full IMS design (e.g.
local monitoring on a Client system with authentication / supervision performed by a remote Host). The keystroke analysis system was considerably enhanced to operate as a transparent background task on a monitored PC station, with impostor detection being performed by an independent machine. The description in this chapter essentially details the main features of the system, highlighting the aspects of the IMS design that have been realised (which also include system configuration auditing and anti-virus protection). The demonstrator provides a good basis for showing how an intrusion monitoring security system would operate in practice.

Returning to a theoretical level, chapter 10 discusses how a full intrusion monitoring system implementation could be integrated into a wider framework for healthcare security, with systems involving inter-HCE communications. The suitability of cryptographic protection techniques is examined, along with a discussion of how Trusted Third Party (TTP) systems could be used to provide an independent assurance mechanism. The applicability of these technologies to the healthcare community is illustrated using a further example scenario to support the main points raised. This aspect of the discussion represents the integration of the research into an overall protection framework and is provided as a pointer to possible future work.

Finally, chapter 11 presents the main conclusions arising from the entire research programme, highlighting the principal achievements and limitations of the work, along with suggestions for potential further development.
The thesis also includes a number of appendices which contain a variety of additional information in support of the main discussion (including a number of published papers from the research programme).
CHAPTER 2

The need for Security in Healthcare
2.1 An overview of Information Security

A generally accepted view of information security is that it is centred around the preservation of the key factors described below (Commission of European Communities 1991).

- **Confidentiality**
  
  This refers to the prevention of unauthorised information disclosure. All access to data must be restricted to authorised users who have a legitimate "need to know". The seriousness of disclosure may often be dictated by whether it occurs to an unauthorised member of the same organisation or a total outsider (with the consequences from the latter being potentially more severe).

- **Integrity**
  
  This refers to the prevention of unauthorised modification of information. There is an implicit requirement for users to be able to trust the system and be confident that the same information can be retrieved as was originally entered. Integrity may potentially be compromised as a result of accidental error or malicious activity.

- **Availability**
  
  This identifies a requirement for data and systems to be accessible and usable (by authorised users) when and wherever they are required. This necessitates both the prevention of unauthorised withholding of information or resources, as well
Chapter 2: The need for Security in Healthcare

as adequate safeguards against system failure. The seriousness of any denial of service will, in most cases, increase depending upon the period of unavailability.

The requirements above may be compromised by a variety of threats to the system which, if realised, could result in security breaches. At a high level, these may be grouped into three main categories, resulting from a variety of accidental or deliberate acts against which systems must be protected:

- physical threats (e.g. fire, flood, building or power failure);
- equipment threats (e.g. CPU, network or storage failure);
- human threats (e.g. design or operator errors, misuse of resources, various types of malicious damage).

It is recognised that, without adequate security provision, the above threats may lead to a number of undesirable consequences, or impacts (Davey 1991):

- disruption of activities;
- embarrassment or loss of business goodwill;
- breach of personal privacy or commercial confidentiality;
- failure to meet legal obligations;
- financial loss;
- threat to personal safety.
At one time these issues could have been adequately addressed by mainly physical means of protection (such as locating equipment in locked rooms and restricting physical access). With system access only possible from within this environment, many security problems were essentially resolved. However, such centralised computer centres are now being replaced by networked mini and personal computers and the use of information systems has moved well beyond the stage where physical security mechanisms alone will suffice. The advent of widespread data communication networks, increased end-user access and multimedia systems has meant that the focus must alter significantly. Physical security is still an issue, of course, but it is now just one of several perspectives from which protection must be considered (which now include logical security, with issues such as user authentication and encryption, and procedural / personnel security, covering more staff or policy related measures) - all of which limit vulnerability to threats and thereby reduce the resultant impacts. These different views of security are illustrated in figure 2.1 below.

![Perspectives on information system protection](image-url)
All of the factors mentioned may potentially pertain to the healthcare scenario and, as such, the specific requirements of the healthcare environment will now be considered in more detail.

2.2 Current trends in European healthcare and informatics

This section attempts to provide an overview of the current situation in the European healthcare community, looking firstly at the demands placed upon health services in general and then the ways in which information technology increasingly has a role to play.

2.2.1 Healthcare in the European Union

The most recent overall assessment of health services at a European level (De Moor et al. 1994), based upon 1990 statistics, indicates that around 6.6 million people are employed in the healthcare field, distributed as shown below:

- 800,000 doctors;
- 156,000 dentists;
- 200,000 pharmacists;
- 1.6 million nurses;
- 3.86 million “other” personnel in supporting roles such as administration and information technology.

There are around 15,000 hospitals across Europe, providing a total of some 2.6 million beds (a figure which obviously does not take into consideration the number of smaller healthcare
practices) and of the 344 million European inhabitants, each person visits a doctor an average of five times per year.

It is possible to cite a number of trends in respect of healthcare demand and provision (Barber 1991a):

- an increasing demand for health services, particularly from the increasing proportion of elderly patients for whom care is more costly;

- an increasing ability on the part of HCEs to provide life-saving and life-enhancing treatments;

- an increasing expectancy that services will be provided, coupled with increased public awareness of the services that can be successfully delivered.

The healthcare system must obviously attempt to meet these demands. However, various factors may serve as handicaps in this respect, including funding constraints, a reduced number of individuals entering health services (partially due to a general reduction in the young population) and an increasingly unequal distribution of medical expertise within the community (Barber 1991a).

One means of coping with the increased requirements has been through the use of technology to improve the speed and efficiency of healthcare operations. As such, information systems are now routinely used throughout the community.
2.2.2 Use of Information Technology

When computers were first introduced into healthcare they were initially used "at a distance" from the patient (e.g. for HCE administration purposes). However, the situation has since altered dramatically, and modern establishments now utilise a wide variety of equipment, ranging from standalone PCs to minicomputers and mainframe systems, with increasing volumes of data transmitted between different locations. As such, IT now affects most areas of HCE operation, from a continuing role in administration through to patient care, with computers now used to directly control medical equipment linked to patients and provide information or advice to clinicians.

In the UK alone, the annual IT expenditure of the National Health Service is several hundred million pounds (The Guardian 1994) and similar levels of funding are likely to be observed across Europe. As a consequence, the range of current and emerging uses of information systems is enormous and some illustrative examples are listed below (all of which also pose concerns in terms of security):

- interconnection of computer systems and institutions;
- increasing storage of highly sensitive data (e.g. genetics, contagious and incurable diseases);
- widespread use of personal computers (with the focus on the end-user);
- use of clinical expert systems (Riddington et al. 1994; Khoor at al. 1994);
- computerisation of primary care practices;
- the development of telemedicine;
Chapter 2: The need for Security in Healthcare

- mobile computing;
- introduction and use of smart cards for patient data storage (Pangalos 1992).

In view of these points it should not be too surprising that the increasing pervasion of technology has had a significant effect upon standard healthcare practices, leading to a number of further trends (Barber 1991b):

- more medical systems, networks and users - accompanied by a proportional increase in the chance of errors;

- an increased number of applications directly relating to clinical care, with a significant proportion in safety critical areas;

- increased information sharing and access from other HCEs;

- widespread reliance upon the availability of systems and the correctness of output. Healthcare staff at all levels will now be less experienced in working without access to and support from information systems. As such, there is significantly less likelihood of staff still being able to handle work manually within the necessary timescales if systems fail.

General evidence of these trends and the dependence upon IT is provided by a reasonably recent survey involving 120 medical staff (Al-Hajjaj and Bamgboye 1992). This revealed
that some 75% of those questioned considered computers to be either important or essential in modern healthcare practice.

As a result of these considerations, information technology may now be regarded as a central, and vital, asset in many aspects of HCE operation (Abbott 1992), representing a significant investment in terms of both finance and information. As such, one would reasonably expect that sufficient measures would be employed to safeguard healthcare systems. However, in practice this has not really been the case. For example, whilst the UK National Health Service has had a formal IT strategy since 1986 (NHS 1986) it is only in recent years that security and data protection issues have been properly addressed in a similar manner.

2.3 Security requirements in healthcare

The widespread use and reliance upon information technology can be cited as the first major factor dictating a requirement for security in healthcare establishments. Whilst the increased use of IT in healthcare should be beneficial at all levels, a more pessimistic view is that this means there is simply more potential for something to go wrong. For example, a computer-based healthcare records system is advantageous in that it allows more information to be collected and stored than its paper-based counterpart. However, this in turn serves to make the computer system a more attractive target and it is likely that a security breach would result in the disclosure of information about a large number of patients.
A second consideration is that protection requirements have been recognised on both national and European levels, leading to increasing requirements to comply with various legislation (as will be briefly discussed in section 2.5.1).

A final influence is the actual evidence of security incidents that have occurred within healthcare. A number of specific examples have been identified in recent years (Barber et al. 1993a), including:

- destruction of computing facilities by hurricane winds;
- loss of emergency calls by ambulance software;
- errors in the calculation of radiation treatment plans;
- errors in financial systems.

In addition, the most recent Audit Commission survey on computer abuse in the United Kingdom (Audit Commission 1994) indicated that 24% of the reported incidents from respondent organisations occurred in healthcare establishments (a point which is examined in more detail in chapter 6). All such issues are bound to have some impact on public trust in the HCEs involved and, in a worst case, a really serious security breach could completely undermine the reputation of an establishment.

The combination of these points effectively highlights why the issue of information security may now be considered to be of increased importance in health systems.
Chapter 2: The need for Security in Healthcare

Luckily, whilst the use of technology might increase the threats associated with health data, it can also offer the means to compensate through the provision of better protection. However, Bakker (1991) believes that the new threats are not properly counter-balanced by new security measures. Whilst this will certainly be true in some respects, it is also often the case that available protection technologies are not utilised. This point is evidenced by the fact that a large number of operational health systems were developed without security considerations in mind. There are a number of potential reasons why such omissions / oversights may have occurred:

- the view that because healthcare “means well” (and therefore should not represent a threat to anyone), information should not require significant protection;

- the nature of the information stored is secondary to its use in the provision of care;

- security represents an overhead (in terms of both finance and effort) that does not directly contribute to the primary objectives of the HCE. As with most other organisations, HCEs have to contend with finite levels of resources and increasing costs. As a result, expenditure is prioritised such that clinical services (i.e. those in direct contact with patients and advantageous to a large population) will generally obtain funding more easily than support services. The frequent reports of shortages in healthcare (e.g. in terms of waiting lists for beds and treatment) indicate the level to which these resources are often restricted.
Hence investments to improve security may be considered somewhat secondary and will normally only be approved if it can be demonstrated that the benefits to the HCE will be of greater importance than if the money was directed at the addition or enhancement of clinical services (AIM SEISMED 1993a).

Whatever the reason, there is now a requirement for security to be “bolted on” in many cases. However, it is important to carefully consider how this can best be achieved, especially given that healthcare is fundamentally different from other sectors in terms of acceptable security.

2.4 Special factors of the healthcare environment

The fact that security needs can be identified in many other information technology domains raises the question of why healthcare should demand special consideration. Curiously, an AIM Working Conference on Data Protection and Confidentiality (Commission of European Communities 1991b) suggested that it was “probably not possible to draw a distinction between medical requirements and needs and those from other sectors or the general domain”. Whilst this may be true from the perspective that many protection methods appropriate to other domains will also be applicable in HCEs, it tends to overlook the fact that HCEs may have some unique requirements and constraints in respect of security. Some of these may be explained in relation to the issues of confidentiality, integrity and availability that were highlighted in section 2.1, along with more general observations relating to the nature of the environment and staff culture, which will be discussed later.
In the majority of other sectors where obvious requirements for information security may be perceived (e.g. in police, military or government systems), it is normally confidentiality that is regarded as the over-riding concern. In healthcare, however, the issues of integrity and availability are potentially more serious as problems in these respects could, in a worst case scenario, result in the most severe consequence of all, namely loss of life. Each of the issues will now be considered in more detail in the healthcare context.

2.4.1 Confidentiality

Confidentiality may be viewed as more critical in HCEs than in many other sectors as there are several aspects to the problem that must be considered. These result from the handling of potentially sensitive data relating to both patients and the institution.

Firstly, the principle of confidentiality is fundamental to medical practice in that it provides an assurance to the patient that discussions with the doctor will not be divulged to others. This attitude stems directly from the Hippocratic oath; the oath of ethical professional behaviour that is regarded as the basic duty of Healthcare Professionals (HCPs) in almost all European countries:

"Whatever, in connection with my professional practice, or not in connection with it, I see or hear, in the life of men, which ought not to be spoken of abroad, I will not divulge, as reckoning that all such should be kept secret." - Hippocratic Oath.

This is central to maintaining a necessary relationship of trust between patient and practitioner and the moral responsibilities of HCPs should obviously remain the same
regardless of whether records are paper or computer based. National and European data protection legislation will also be relevant in this context, as described later in section 2.5.1.

The unauthorised disclosure of patient healthcare information has the potential to lead to a number of undesirable consequences for the affected individual. At the most basic level it will represent the infringement of the patients general rights to privacy. However, more serious or damaging effects may be:

- potential to cause embarrassment;
- potential to cause discrimination, prejudice or even social ostracism;
- potential to invite blackmail or bribery.

Particularly relevant in this context will be especially sensitive classes of data, examples of which would include any of the following:

- sexually transmitted diseases (STD);
- mental health;
- drug addiction or alcoholism;
- child abuse;
- adolescent healthcare (i.e. that sought without parental knowledge).

Maintenance of patient confidentiality is also in the interest of the HCE in order to avoid the risks of litigation and adverse publicity.
There are also various types of information that the HCE itself may wish to remain confidential. For example, there may be commercial or organisational implications if business data relating to any of the following were to become generally known (especially in the context of the more performance-related HCE environments now emerging):

- financial constraints;
- resource shortages;
- undesirable performance indicators (e.g. relating to staff or the organisation).

However, despite the considerations identified above, confidentiality may be regarded as a somewhat less serious issue in many health systems (particularly those with direct links to care delivery) when compared to the demands for data integrity and availability.

2.4.2 Integrity

The loss of data integrity is potentially the worst scenario in healthcare as it could most easily cause actual harm to the patients (i.e. as a result of decisions and treatment being based upon incorrect information).

Errors can occur in healthcare records for a number of reasons, but in general the requirement for data integrity in healthcare must address two aspects: the preservation of accuracy and the prevention of unauthorised modification. The first point recognises, for example, that under current practice the same data is often duplicated in independent systems, leading to a risk of inconsistency. Mechanisms are, therefore, required to ensure
that modifications to data in one system are reflected in all others. The implication of the second point is that unsuspected, unauthorised modifications could lead to misinformed decisions which, in a clinical context, could result in serious harm to patients.

Examples of integrity failings in the UK health service include several widely publicised cases where computer-controlled radiotherapy equipment administered incorrect radiation doses to cancer patients. Whilst the root cause is normally human error, the involvement of IT is highlighted in all cases. The North Staffordshire Royal Infirmary and the Royal Devon and Exeter Hospital were amongst the establishments involved (The Guardian 1992; Sunday Times 1993).

Roger France (1994) identifies an additional consideration influencing the requirement for integrity, citing that the handling of health data often has to be done practically “on-line” with little or no time for further checking. This practice allows data to be made available for care purposes as soon as possible (e.g. there will frequently be no delay between the receipt and utilisation of laboratory results data). It is obviously important that the data obtained is correct in such circumstances.

Additional considerations here might be the risk of legal action being taken against clinicians or the HCE as a result of its failure to protect the information (e.g. representing a breach of data protection legislation).
2.4.3 Availability

The availability of healthcare systems and data may be important at various levels. In general, the unavailability of even the most mundane healthcare systems (e.g. patient appointments) could result in inconvenience, delays and the like. It is fair to say that most HCPs expect systems to be available on demand, and a significant proportion (particularly those directly relating to care delivery) will be required to be operational 24 hours a day (see chapter 3).

In terms of care delivery and decision making, healthcare professionals are not only reliant upon their own skills, but also upon the information that is at their disposal. Whilst there is an expectation that competent and experienced HCPs would be able to cope in many situations even if patient records were unavailable, most would still prefer access if record were known to exist (in case additional beneficial information was available). The unavailability of data (e.g. patient medical histories or diagnoses) could realistically reduce the quality and effectiveness of treatment, given that decisions would be made on a less informed basis (Barber et al. 1992). Whilst, in most cases, unavailability would be unlikely to be "life threatening", such scenarios are not inconceivable (e.g. a system controlling the automatic administration of drugs could be seriously affected if data was unavailable for even a short period).

A number of general observations can be made regarding the availability of healthcare data. Firstly, the impact of unavailability will usually be most critical in the case of recently recorded data (e.g. that relating to current patient cases), and will become less significant for historical data.
A second point is that the destruction of information, although significant, should normally be less serious than a loss of integrity. The reasoning here is that it is more likely to be recognised, allowing short-term alternative arrangements to be adopted.

Finally, an important consideration should be that security measures themselves do not unnecessarily impede or prevent HCPs from getting access to required information (especially in vital emergency or exception cases). This point leads into a more general discussion of the need for convenience, which is presented in the next section.

2.4.4 General issues

A number of further factors can be cited as being relatively special in HCEs. Principal amongst these are the open environment, the need for convenience, the staff culture and financial constraints.

The generally open environment in HCEs means that many physical controls are not workable and the high degree of public access means that unfamiliar faces will be commonplace in most areas. As such, anyone who looks appropriate is unlikely to be challenged, whatever their activity (as evidenced by the recent incidents of babies being kidnapped from maternity wards (The Times 1994a; The Times 1994b)). At the same time there is a necessity for information systems to be inter-mingled in many public areas (e.g. on wards or in reception / waiting areas), leaving them potentially vulnerable to abuse. However, it is obvious that the openness of the environment must largely be maintained and, therefore, the security side must be either compromised or addressed differently.
The issue of convenience will influence the types of security that are appropriate to, and will be tolerated within, a HCE. Young (1991) cites that there are often problems enough entailed in trying to get HCPs to use information systems in the first place (as a result of system designers ignoring the clinical environment and the ways in which HCPs are motivated) and, as such, the addition of cumbersome or restrictive protection measures would only be likely to worsen the situation. For example, effects on staff might include demotivation and reduced efficiency, whilst at an organisational level operational costs could increase as a result of tasks taking longer to perform.

In some contexts this significantly limits the types of security that are appropriate. In terms of system security it is particularly difficult at present to utilise strong methods of security whilst still maintaining a convenient and user-friendly environment. The transparency of security mechanisms is one of the key issues promoted later in this investigation.

The issue of staff culture (i.e. the typical attitudes and behaviour of members or groups within an organisation) highlights the fact that there are discrepancies in the need for security as perceived by technologists and as seen by HCPs. This comes back to the earlier mentioned point that healthcare is seen as being good-intentioned and, therefore, a need to have security is essentially contradictory to this view. In addition, healthcare users are generally no different to those in other sectors in terms of a tendency to regard security as "someone else's problem" and, hence, often have little appreciation of the main issues. However, all healthcare staff involved in the development, operation, maintenance and use of information systems should be responsible to some degree. It has been observed that security is a human issue (Warren and Gaunt 1993) and there is consequently a definite need
to move towards a more security conscious culture in HCEs (where security ideally becomes an ever-present background consideration for all system users). The first step here must be the establishment and adoption of a suitable high-level security policy (Katsikas and Gritzalis 1994) that can provide a common staff reference point, specifying the means by which the HCE should operate in order to preserve security. However, the other factors mentioned in this section will obviously limit the extent to which this can be achieved.

Finally, as previously identified in section 2.3, the issue of finance will influence the viability of security in healthcare. As an illustration of the financial constraints that may be faced, European Commission (De Moor et al. 1994) statistics indicate that expenditure on informatics represents only 0.4% of the yearly running costs of a hospital (with the majority taken up by supplies (26%) and personnel (68%) costs).

2.5 Addressing healthcare security in Europe

Whilst possibly not regarded as a top priority in the past, the need for security has nevertheless been recognised in many HCEs and steps taken to address the issue. However, this fragmented approach often leads to subjective views of what protection is appropriate and hence inconsistency between similar establishments and systems.

As a result, the need for improved methods is widely recognised in the European healthcare community and various steps have been taken to address the problems. Several key aspects are described in the sections that follow, one of which was closely related to the research programme.
2.5.1 An overview of previous European initiatives

One of the key European initiatives in respect to the use of information technology in healthcare has been the AIM (Advanced Informatics in Medicine) programme. The objective of AIM was to increase harmony and cohesion in the European healthcare community, whilst at the same time improving the quality and cost-effectiveness of medical care. The need for safe and secure healthcare services was one of the obvious areas that the programme addressed.

A notable step in this respect was the development of what are known as the “Six safety first principles of health information systems” (Barber 1991a). These were specified by the AIM Requirements Board as a basis for future development, testing, operation and maintenance of healthcare information systems, and are intended to encourage the use of such systems within Europe (with conformance to the principles being seen as a way to increase the confidence of both HCPs and the public). The six principles are listed below, along with a brief statement of the purpose in each case.

1. Safe environment for patients and users

   To ensure that no-one is harmed by the operation or non-operation of systems.
   
   This includes the issues of quality control and assessment.

2. Secure environment for patients, users and others

   To ensure that information is not lost, corrupted or made available to others.
   
   These points, of course, relate closely to the previously mentioned issues of availability, integrity and confidentiality.
3. Convenient environment for users

To ensure that systems can be used easily and effectively within HCEs.

4. Legally satisfactory environment across Europe for users and suppliers

To ensure that legal, ethical and professional responsibilities are harmonised within the healthcare community.

5. Legal protection of software products

To ensure that software products are protected against unlicensed use and thereby foster/promote the European market for health informatics systems.

6. Multi-lingual systems

To facilitate the spread of systems throughout Europe and avoid errors resulting from inadequate understanding of local languages.

Whilst the second principle can be most clearly related to addressing healthcare security requirements, all of the points are important (to some extent) in ensuring the overall secure and error-free operation of healthcare environments and systems.

In addition to these principles, there is also a variety of national and international legislation within Europe that either directly or indirectly relates to healthcare and thus influences the requirements for protection.
Chapter 2: The need for Security in Healthcare

The most notable legislation in the UK currently includes:

- Data Protection Act (1984);

Whilst both of these acts are targeted at the IT community in general, they also apply in the healthcare scenario (although, of course, the Computer Misuse Act serves more as a form of deterrent, providing a legal recourse for HCEs in cases of abuse, rather than stipulating a requirement to protect systems). Similar types of national legislation are reflected in most other European countries (Lobato de Faria 1992).

At the European level, the most significant legislation is a draft directive on data protection (Walden 1990). This lays down very comprehensive rules for the handling of data and seeks to cover every situation in which the processing of personal data may involve a risk to the data subject (relating to data in manual or automated systems, in both the private and public sector).

The Council of Europe has also been active in the areas of data protection and medical information systems, and has made several significant contributions during the past decade or so. These principally include the following:

- Convention 108 for the Protection of Individuals with Regard to Automatic Processing of Personal Data (1981);
Chapter 2: The need for Security in Healthcare

- Recommendation No R (81) on regulations for automated medical data banks (1983);
- Recommendation No R (87) 23 on Hospital Information Systems (1987);
- Draft Recommendation on the communication of health information in hospitals: ethical and legal issues (1992);

Useful summaries covering the main issues raised in these regulations and their applicability to the healthcare community have been produced by Lobato de Faria (1992) and Duerinckx (1993). However, it can be observed that most limit their concern to the areas of data protection and confidentiality rather than addressing the wider security field.

However, despite all of the aspects discussed, there are still practical problems in the actual realisation of healthcare security. Firstly, much of the European material is not formally binding and, as such, organisations are not legally obliged to follow the recommendations (which can again lead to inconsistency in terms of protection or, at worst, the guidance being ignored). More significantly, much of the material is of a very high level nature, providing general statements of “good practice” rather than more solid recommendations that can be followed. For example, the eighth principle of the UK Data Protection Act states that:

"Appropriate security measures shall be taken against all unauthorised access to, or alteration, disclosure or destruction of, personal data and against accidental loss or destruction of personal data."

36
However, there is only a very minimal amount of accompanying material to state what issues should be considered and nothing relating to actual security measures. As such, it is observed by Barber (1991b) that most HCEs would be unlikely to pass even a basic data protection audit.

Similarly, the six safety first principles, whilst intentionally presented in only general terms (to allow them to be set apart from specific computer systems and problems), consequently only state what should be done rather than how one should go about it.

As a result, more practical recommendations are required by which HCEs can actually identify and address their security needs. A significant European attempt to realise this objective is the AIM SEISMED project, which is introduced in the next section.

2.5.2 The AIM SEISMED project

The initial stages of this research were closely linked to work being conducted under the banner of the Commission of European Communities SEISMED (Secure Environment for Information Systems in Medicine) project, part of the aforementioned AIM programme.

The objective of SEISMED was to provide practical security advice and guidance to all members of the healthcare community who are involved in the management, development, operation or maintenance of information systems. The eventual aim of the project was to establish a consistent framework for the protection of medical data across the European Union (AIM SEISMED 1991).
The project was structured in three main phases as described below.

1. The identification of information security requirements within the European medical community (including investigation of the protection practices currently in use).

2. The development of guidelines by which secure systems may be designed and the security of existing systems may be enhanced.

3. The implementation and evaluation of the above guidelines in four European healthcare establishments selected as project "Reference Centres" to ensure their practicality and applicability.

Work on the project commenced at the beginning of 1992 with an original duration of three years, but this was subsequently been extended for a further 6 months (until mid-1995).

A total of 14 workpackages were established within the project, each addressing a separate aspect of healthcare security. The contributing partners were comprised from establishments across the community. Principal amongst these, as regarded getting representative European medical input and opinions, were the aforementioned Reference Centres. These establishments were located as follows:

- Plymouth Health Authority (United Kingdom);
- The Royal London Hospital NHS Trust (United Kingdom);
Chapter 2: The need for Security in Healthcare

- Leiden University Hospital (The Netherlands);
- GEN Hospital Cantonal Universitaire de Geneve (Switzerland);
- Institute for Clinical and Experimental Medicine (Czech Republic).

The reference centres are indicated on the map in figure 2.2, which also highlights those countries directly involved in SEISMED and additional countries that participated in a healthcare security survey conducted by the project (the results of which are described in the next chapter).

Fig. 2.2: SEISMED Reference Centres, partners and collaborators
This research programme was primarily linked to the SEISMED project through workpackage SP07, which dealt with *Security in Existing Systems*. The scope of the workpackage was to produce a comprehensive set of recommendations for the addition (or enhancement) of security in operational healthcare systems and environments. More formally, the three main objectives of the workpackage, as stated in the original Technical Annex of the project, were as listed below.

1. To produce guidelines on the level of protection that should be attached to existing operational healthcare systems.

2. To produce guidelines as to how the appropriate level of security in existing systems may be achieved.

3. To revise the approach based on Reference Centre feedback.

A significant proportion of the work described in this report was conducted to assist in the fulfilment of these objectives.

### 2.6 Conclusions

This chapter has shown that whilst a significant requirement for information system security exists in healthcare, the nature of the environment itself imposes some constraints upon what is acceptable.
Chapter 2: The need for Security in Healthcare

Having identified the basic concerns, it is necessary to show that many existing European systems are actually deficient in terms of the protection provided. This issue is addressed in the next chapter, with the discussion of a survey of existing security practice.
CHAPTER 3

A Survey of European Healthcare Establishments
3.1 Overview and objectives

A broad picture of healthcare security within the European Community was obtained as a result of a questionnaire study conducted within the SEISMED project during 1992 (AIM SEISMED 1995a). The objective of this was to identify and assess current HCE practices and attitudes regarding information systems security, which could then be used as a foundation for further work within the project.

Whilst the questionnaire used in the study was devised prior to the commencement of this research programme, the results of the survey were available for the purpose of assessing existing security weaknesses (or inconsistencies) and deciding the appropriate means of resolving the problems. As such, the procedure and relevant findings of the survey are summarised in the sections that follow (whilst a copy of the relevant sections of the questionnaire itself can be found in appendix A).

In order to obtain a true European perspective, questionnaires were distributed to healthcare establishments in the following countries:

- Belgium  
- Denmark  
- France  
- Germany  
- Greece  
- Ireland  
- Netherlands  
- Portugal  
- Sweden  
- Switzerland  
- United Kingdom
Chapter 3: A Survey of European Healthcare Establishments

The survey aimed to elicit responses on a broad range of security-related issues including types of operational system and data, awareness of the need for security, awareness of relevant protection legislation and current technological environments. This was accomplished using a total of 56 questions, 18 of which were relevant to the assessment of existing systems. It was sent to larger institutions, such as general hospitals, as opposed to the primary care establishments of general practitioners.

Usable responses were received from 75 establishments, with respondents representing all major categories of healthcare professional and possessing varying levels of both professional and security-related experience. This response base was relatively small and could not be considered to provide a representative sample across the community (e.g. only one response was obtained from each of France, Portugal and Sweden, whilst 43 came from the United Kingdom). However, the results do serve to give a high-level view of the current situation and it was possible to extract a substantial amount of information relevant to existing systems, allowing an overview of current security practice to be gained.

3.2 Survey findings

The following sections of the survey enabled information to be obtained regarding the types and level of security currently found in existing systems:

- Physical Security Details of your System;
- Design Security Details of your System.
Responses were analysed to allow information to be extracted pertaining to four key areas of interest:

1. system configurations;
2. main application areas;
3. existing security measures;
4. security problems encountered;

The survey considered these issues in significant detail and a full reproduction of the results would be outside the scope of this report. The analysis that follows is based on a summary of the most relevant statistics, and should be sufficient to allow an appreciation of the main problems and requirements.

3.2.1 System Configurations

As one would expect, a wide variety of different system architectures are employed across Europe. As a basic distinction, the survey gave separate consideration to minicomputers / mainframes and personal computers.

The majority of sites claimed at least one mini or mainframe based system, with over 16 different hardware and operating system platforms being listed. Most HCEs have equipment from more than one supplier and, therefore, whilst a dominant architecture could be identified (DEC VAX being used by just over 70% of sites), it was normally used in conjunction with a number of other systems.
In the personal computer field things are slightly more clear cut, with IBM compatible machines and MS-DOS accounting for around 75% of responses. However, the remainder of the small systems were based upon a wide range of other hardware and operating systems.

In both of these cases, the fact that no system is universally accepted precludes the option of specifying system-specific standards for protection.

Communications feature significantly in most environments, with 78% of systems being part of a network. Of these, 92% of systems are part of a local area configuration and 55% are connected to wide area networks. The latter case illustrates a substantial requirement to transmit and receive information from outside a single establishment.

3.2.2 Main Applications

Respondents were asked to indicate which of six named applications were present in their environments, along with any other significant systems. The results of this are summarised in figure 3.1 and it is clear that the named applications can be found in the majority of environments. It would, therefore, appear advantageous to devise appropriate protection profiles for these types of system that could then be applied in any scenario. However, this approach in itself would be insufficient as it overlooks two key factors. Firstly, there are still a significant number of “other” application types (25 identified in all, as listed in Appendix A) for which security may also be required and, secondly, these profiles would not take into consideration the underlying hardware / software systems that support the application.
In terms of the security requirements associated with these systems, it can be assumed that confidentiality will be required in many cases (this will depend largely upon the data involved, making it hard to generalise), and that integrity will be expected regardless. As a result, the survey did not attempt to collect any opinions on these issues. It does, however, relate information pertaining to the issue of availability, and the charts in Figure 3.2 give an overview of the performance expectations associated with healthcare systems in general.

The first chart indicates the typical period of time during a day that systems are expected to remain operational. This reveals that a clear majority of systems are required to be available most of the time. The second chart indicates the length of time that is considered acceptable for a system to satisfy an information request (in average cases - the times could
be expected to be less in emergencies). These figures underline the importance that is attached to availability (substantiating the observations made in chapter 2), and indicate that any denial of service could have serious consequences.

![Planned hours of operation](image)

![Acceptable Information delay](image)

**Fig. 3.2 : General availability expectations**

### 3.2.3 Existing Security

Bearing in mind the expectation that security in existing systems would be somewhat weak and inconsistent across the community, the survey asked respondents to indicate which types of security were present in their systems. Figure 3.3 illustrates the relative adoption of six key categories of protection (it should be noted that the percentages in the chart serve to compare the acceptance of the different measures rather than indicate the proportion of cases in which they have been implemented).
Chapter 3: A Survey of European Healthcare Establishments

Fig. 3.3: Relative adoption of security techniques

In a reasonably secure system, one would expect that all of these areas (with the exception of encryption and possibly audit trails) would be given roughly equal consideration. However, even at this high level it is apparent that some areas (e.g. disaster recovery) are given substantially less attention than others. Whilst it could be argued that some systems genuinely demand less in terms of disaster recovery than they do authentication, it should be noted that these proportions are maintained across all the main systems considered. Using the same example, logic would seem to indicate that if a system or data is worth protecting against unauthorised use then it should almost certainly be sensitive / valuable enough to safeguard against damage or possible destruction as well.

Inconsistencies become even more apparent when the individual areas are examined in more detail, and the survey highlights significant discrepancies in the types of countermeasure
employed to achieve each aim. This may be illustrated by considering the variety of mechanisms used to accomplish the most widely accepted goal, namely Authentication.

Fig. 3.4: Methods of authentication

It can be seen from figure 3.4 that there appears to be little standardisation in the approaches taken. A further observation is that very few cases appear to use anything other than variations on the simple password, the potential weaknesses of which are well documented (Jobusch and Oldehoeft 1989). Similar criticisms can be levelled at breakdowns of the other groups from figure 3.4.
3.2.4 Security Problems

The final area of interest in the results was data relating to experience of security problems. Twelve types of incident were considered and respondents were asked to indicate the frequency with which these had occurred (if at all). The cumulative results are presented in figure 3.5.

![Figure 3.5: Security Incidents in European HCEs](image)

This range of incidents provides further justification for the need to enhance the existing protection in many systems. It is important to note that the frequency of occurrence should not be regarded as the sole determinant of protection; even a one-off breach may have serious consequences. Unfortunately, the survey gives no indication of the precise impacts that may have resulted from these incidents. Nevertheless, the figures still represent a significant number of incidents that would at the very least have caused inconvenience to
the HCEs involved. In addition, the results do not allow a correlation to be made between the problems encountered and the level of existing security measures (which may have given an indication of which types of security were most effective).

3.3 Conclusions

Whilst the small number of responses received meant that the results could not be considered properly representative, they were nevertheless felt to be indicative of the current European healthcare situation.

The first overall conclusion that can be drawn from the survey findings is that they confirm the increased use and importance of IT in healthcare. They also reveal significant variety in the types of medical system currently in use and discrepancies in the existing protection being afforded (including likely inconsistency in the levels provided even to systems of a similar nature). This provides reasonable grounds for patients to assume that treatment would generally be safer in those countries / establishments where security and protection issues are properly addressed.

These points underline the requirement to address the security implications more seriously and provide the justification for much of the further work described in this thesis. As a first step, there is a requirement for a formalised approach to security enhancement that may be applied on a general level. This will be the main topic of the next chapter.
CHAPTER 4

A European Standard for Healthcare Data Security
4.1 Approaches to securing existing systems

The results of the SEISMED survey in the previous chapter underlined the fact that information systems security in European HCEs is currently anything but standardised. It is considered that, in many cases, the disparity of security measures has resulted from an overall lack of appropriate standards and guidance. More generally, healthcare practitioners are often so preoccupied with their professional activities that they are not aware of computer security concepts or have only a token appreciation of them. As a result, HCEs are unclear over both general protection issues and the level that they should aim for. It was the objective of the SEISMED project and, in particular, the SP07 workpackage to remedy this situation by establishing the methods by which existing systems security measures could be added (or enhanced) and the systems themselves thereby brought up to an appropriate standard.

Discussion within the SEISMED consortium identified two approaches by which security of existing systems could be addressed, as shown in figure 4.1.

Fig. 4.1: Approaches to existing systems security
As a first stage of improvement, the most appropriate strategy was considered to be the definition of *baseline* recommendations for security, to provide a common foundation for all HCEs. As such, it was proposed that the baseline guidelines should be considered by HCEs as the basis of protecting all systems. The second approach, (i.e. that of the protection profiles) would then be utilised in especially sensitive scenarios.

This chapter will focus upon the new guideline sets that have been established, whilst the idea of protection profiles (and an accompanying methodology that has been developed) will be discussed in chapter 5.

### 4.2 Baseline security for healthcare systems

The concept of establishing a healthcare security "baseline" immediately raises the question of what level of security should be specified. Whilst various guidelines and standards for IT security have previously been developed, none have specifically targeted the needs of the medical community at a European level. As highlighted in chapter 2, the nature of the healthcare environment, with the inherent requirements to maintain patient safety and confidentiality, demands that protection should generally be higher than in many other domains. As a result, the security requirements extend beyond the levels proposed by many existing standards. At the same time, it has previously been cited (Louwerse 1993) that the stronger levels of security attainable under many existing security classification schemes (e.g. the US "Orange Book" (Department of Defence 1985) and European ITSEC (Commission of European Communities 1991a)) are not directly applicable to healthcare due to:
complex systems (e.g. a wide variety of data, with differing levels of sensitivity, 
may be found within a single healthcare system);

low security awareness (e.g. low funding and generally low acceptance of 
“strong” security measures means that recommending them is only likely to 
increase the gap between theory and practice).

As a result, the new baseline recommendations were developed to satisfy the following 
aims, whilst still attempting to remain practical for healthcare implementation :

- to represent a minimum acceptable standard for the security of operational 
  healthcare systems and their associated environments;

- to be usable by all HCEs and staff within Europe;

- to allow a straightforward means of validating existing systems security to ensure 
  compliance.

The development of the resulting guidelines was based upon an interactive approach, in 
close co-operation with the SEISMED Reference Centres and in consultation with other 
independent healthcare professionals. The purpose of this was to ensure that the resulting 
recommendations would be genuinely applicable to the healthcare environment.
From the outset it was established that the recommendations should address more than just the host system in isolation. Indeed, to provide comprehensive protection, several aspects of security must be considered (as previously identified in chapter 2):

- logical / system-based controls;
- physical and environmental protection;
- personnel procedures;
- policy and administration issues.

On the basis of these high level requirements, existing IT security guidelines and standards (NHS Management Executive 1992; CCTA 1993; DTI 1993) were used in conjunction with suggestions from within the project to formulate initial recommendations. These were progressively refined and enhanced over time on the basis of Reference Centre feedback and comments from the independent healthcare personnel. This procedure provided the principal criteria for retention, addition or removal of guideline recommendations.

### 4.3 An Overview of Existing Systems Guidelines

The final *Security Guidelines for Existing Healthcare Systems* (AIM SEISMED 1994a) are grouped under ten key *principles* of protection, representing the main elements governing the security of existing healthcare information systems (having been agreed in detail with the Reference Centres). The principles are denoted by ESP followed by a unique reference code, as listed in table 4.1 below.
Table 4.1: Principles of Existing Systems Security

Each of the principles has a number of associated guidelines. These represent the specific security concepts or countermeasures that should be considered by the HCE to meet the requirements of a given principle. As established earlier, the consideration of existing systems encompasses a very broad range of issues and the overall coverage consequently extends from general concepts to specific technical measures.

The ten protection principles are detailed further below, with description quoted directly from the SEISMED guidelines. In each case the general purpose of the principle is stated, along with a list of the main issues that are covered by the underlying guidelines. A total of 138 guidelines were established and a complete listing of the individual titles (and their applicability to different categories of HCE staff) is given in appendix B.

1. Security Policy & Administration

General Principle
A formal policy will provide clear direction and support for security within the HCE. Policy is formulated from the senior managerial level, with subsequent guidance
provided to all levels of staff. Correct administration of and adherence to the policy should ensure the effectiveness of HCE security controls.

Main issues:

- the need for a security policy;
- policy awareness issues;
- co-ordination and administration of security;
- use of specialist security personnel.

2. Physical & Environmental Security

General Principle
The generally open nature of HCEs and their high degree of public access dictates that physical security measures are a vital first stage of protection to prevent unauthorised access to computing equipment and facilities. Systems must also be safeguarded against a variety of environmental hazards that may adversely affect operation.

Main issues:

- physical access control;
- security of HCE equipment;
- protection against natural disasters;
- environmental controls;
- various procedural measures.

3. Disaster Planning & Recovery

General Principle
The continuous availability of Information Systems is essential to the operation of a modern HCE. It is essential that adequate plans are made to ensure the level of availability needed by the HCE can be maintained in the event of any catastrophe. Recovery of IT systems should be a component of an overall HCE disaster / recovery plan.
Main issues:

- continuity planning (including the development, testing and update of plans);
- fallback arrangements;
- post-disaster procedures and controls.

4. Personnel Security

*General Principle*
The major security weakness of many systems is not the technology but the people involved. Many organisations are extremely vulnerable to threats from their own staff and, as a result, even the most comprehensive technical controls will not guarantee absolute security. There are, however, a number of personnel-related measures that can be introduced to help reduce the risks.

Main issues:

- staff recruitment;
- contractual agreements promoting security;
- security during normal working practices;
- staff appraisal and monitoring;
- termination of employment.

5. Training & Awareness

*General Principle*
Information systems security can only be maintained if all personnel involved in their use know, understand and accept the necessary precautions. Many breaches are the result of incorrect behaviour by general staff who are unaware of security basics. The provision of security training and awareness will make it possible for staff to consider the security implications of their actions and avoid creating unnecessary risks.
Chapter 4: A European Standard for Healthcare Data Security

Main issues:

- the need for general security awareness;
- specific areas that must be addressed (job training, use of information systems);
- recommendations for internal / HCE training and awareness initiatives;
- use of specialist training courses;
- assignment of responsibilities for training.

6. Information Technology Facilities Management

General Principle
A variety of activities can be identified that are related to the normal day-to-day use and administration of information systems. All categories of HCE personnel (management, technical and general users) have responsibilities that must be addressed in order to maintain security in this area.

Main issues:

- system planning and control;
- the importance of maintaining back-ups;
- media controls (e.g. handling, transport and disposal);
- auditing and system monitoring;
- virus controls (e.g. prevention, detection and recovery);
- documentation issues (e.g. availability and control).

7. Authentication & Access Control

General Principle
It is essential that IT systems are protected by comprehensive logical access controls. Access should be guaranteed for legitimate users and denied to all others. All classes of user must be identified and authenticated before any access is granted and further mechanisms must control subsequent reading, writing, modification and deletion of applications and data. There should be no method for bypassing any authentication or
access controls. HCE users are unlikely to be satisfied with controls that intrude upon working practices and chosen schemes should be transparent and convenient in order to gain acceptance.

Main issues:

- requirements for user identification and authentication;
- password issues (e.g. secrecy, selection and changing);
- system and object access restrictions;
- methods of control;
- access in special cases (e.g. system management, third parties, temporary staff).

8. Database Security

General Principle
Database security is concerned with the enforcement of the security policy concerning the disclosure, modification or destruction of a database system's data. Databases are fast becoming very important for HCEs. Over 90% of today's IT systems contain some kind of database and the value of information stored is now widely recognised as a major asset, far more important than any other software. However, databases also introduce additional security concerns (e.g. granularity, inference, aggregation, filtering, journaling etc.) and therefore warrant specific consideration.

Main issues:

- control of medical database software;
- organisation and administration of HCE database systems;
- database operation issues.

9. System Maintenance

General Principle
System maintenance activities merit special consideration given the opportunities that exist to affect the operation of the system. Unauthorised or uncontrolled changes to any aspect of an operational system could potentially compromise security and, in some
cases, endanger life. Maintenance must therefore be carried out in accordance with well-defined procedures.

Main issues:

- controls to prevent unauthorised changes to and upgrades of HCE software, vendor software and operating systems;
- requirements for testing and acceptance.

10. Legislation Compliance

General Principle
Specific levels of protection may be demanded in order to comply with national and European legislative requirements, as well to satisfy internal HCE policy. Whilst the guidelines highlight the most basic requirements, this principle represents an ongoing process which must take account of any new legislation that may be relevant, as well as ensuring compliance with existing standards.

Main issues:

- data protection;
- abuse of information systems;
- prohibition of "pirated" software;
- compliance with internal security standards;
- retention and protection of business records.

4.4 HCE Target Audiences

The new guidelines are intended to provide a common source of reference for European healthcare establishments and are broadly relevant to (and will affect) all categories of personnel. However, it should be evident that many of the specific issues covered will not
be relevant to all HCE staff. As such, the *Security Guidelines for Existing Healthcare Systems* are targeted at three main staff groups (as shown in figure 4.2), with separate guideline sets having been developed for each audience.

### Security Guidelines

![Diagram of Security Guidelines]

- **General HCE Staff** (50 Guidelines)
- **HCE Management** (61 Guidelines)
- **IT & Security Personnel** (122 Guidelines)

**Fig. 4.2**: HCE target audiences

Whilst all three sets draw upon the same core principles, they nevertheless differ dramatically in terms of the type and quantity of information presented. The significance of each audience, along with the anticipated readership and general content of each guideline set, is summarised below.

- The *General* guideline set is aimed at the majority of HCE staff, including clinicians, administrators and general system users. Although the need for security is most likely to be considered in the context of HCPs, it should be remembered that security issues actually apply to all health workers. Many non-care staff (e.g. secretaries and clerks) will also handle sensitive data and make
extensive use of IT systems. Guidelines are presented for user reference during
day-to-day use of HCE information systems, highlighting what they can do to
safeguard security.

• The *Management* set primarily targets the senior decision makers within the
HCE, who will be responsible for defining security policy (although a significant
number of points will also be relevant at department / line management level).
Management will be central in setting the style and standards of operation
within the HCE, but may be unaware of the dangers posed by inattention to the
key security issues. This set is intended to highlight areas in which management
should be directly involved and also improve management security awareness by
explaining / justifying the importance of other more technical guidelines (for
which management approval will be required).

• The *IT & Security Personnel* set is aimed at IT staff, system administrators,
security officers and other support staff who will be most likely to have the
lower level responsibilities for implementing security. Information technology
personnel will occasionally need access to, or come into contact with, health
data as part of systems development, operation and maintenance activities. This
is the most detailed of the subsets and should be a key source of reference for
the implementation and validation of security.

The *Management* and *IT & Security* audiences would also be required to read and observe
the *General* guideline set.
Chapter 4: A European Standard for Healthcare Data Security

A complete description of the individual guidelines and how they apply to each staff group is obviously outside the scope of this thesis. However, as an illustration of how the emphasis is altered for each target audience, the following example guidelines (all taken from the Security Policy & Administration principle and relating to the need to formulate and observe a Security Policy Document) may be considered:

"All users should acquaint themselves with the HCE security policy and observe any general regulations as well as any that may specifically apply to their role or department."

"Written documentation detailing HCE security policy (or a synopsis of the main points) must be available to all personnel. It should contain a clear definition of information security, as well as a clear and unambiguous explanation of the objectives and scope in relation to the HCE. The specific principles and guidelines implemented by the HCE should also be detailed."

"Technical staff should provide relevant expertise to assist management in the formulation of the HCE security policy. They should subsequently acquaint themselves with the policy in full and observe any general regulations as well as any that may specifically apply to their role or department."

4.5 Implementation of the recommendations

It was envisaged that the Security Guidelines for Existing Healthcare Systems would be broadly applicable in any European Healthcare Establishment with existing operational information systems. They will be relevant even where systems are thought to include security provision, so that the level of protection can be validated against the recommendations.
However, given the diverse nature of European healthcare environments and systems, it was
impossible to specify precise guidelines for implementation that would be correct for all
scenarios. Healthcare establishments will differ in terms of both the information systems
used, as well as financial, operational and other constraints that may apply. These issues
will all have bearing on the applicability of the recommendations and the guidelines
therefore concentrated more on describing what aspects of security should be considered
rather than how they may be best implemented (with broad recommendations that should be
compatible, to at least some degree, with the majority of systems and environments).

Despite these attempts to ensure applicability, it was still conceivable that some guidelines
may not be suitable for all systems. In general, the baseline is flexible enough to accept that
some guidelines are not implemented and, as such, implementors can use their discretion in
cases where guidelines are genuinely inappropriate to the environment. However,
recommendations should be followed as closely as possible and in some cases the
implementation of a guideline will rely upon others already being in place (which is made
clear from the guideline context and / or cross-references to other points).

As for the implementation strategy itself, it would obviously be impractical to attempt to
address all of the suggestions at once due to constraints of cost and likely disruption to
services. A phased approach is, therefore, advised in which each principle is considered in
turn to identify the areas in which the HCE / department is currently deficient. The
individual guidelines can then be assessed to determine implementation priorities based upon
local requirements.
Further work within the SEISMED project resulted in the development of the methodology SIM-ETHICS (Security Implementation Methodology - Effective Technical and Human Implementation of Computer based Systems) which may be used to assist with the implementation of these and other SEISMED guidelines (Warren and Gaunt 1993). The methodology is based upon the concept of participational management, using groups of users and managers to carry out a hypothetical implementation of chosen security countermeasures. This provides a means of highlighting any problems which may occur, allowing them to be overcome in advance of the actual implementation. An expert system to support this methodology is currently under development (Warren 1995).

Finally, the Security Guidelines for Existing Healthcare Systems should not be considered in isolation and a number of the other SEISMED guideline deliverables are also relevant in the context of existing systems. These include specific guidelines relating to high-level security policy (AIM SEISMED 1993b), network security (AIM SEISMED 1994b), data encryption (AIM SEISMED 1994c) and system development (AIM SEISMED 1995b).

4.6 Potential Problems

Whilst the new recommendations were intended to provide a simple and straightforward means of addressing healthcare security issues, it was recognised that problems may exist.

Firstly, many establishments may currently be operating with security significantly below the recommended standard and progression to the required level could consequently be a non-trivial task. As mentioned in the discussion of implementation, HCEs may face a number of
constraints that affect their ability to address security requirements. For example, cost (in terms of finance, performance and practicality) will be a significant factor in determining acceptability. Financial cost will be particularly relevant, given the previous observation that expenditure for direct care activities is likely to receive higher priority than security. In addition, organisational constraints will play a role in so far as recommendations will need to integrate with existing practice (or, at least, not conflict too greatly) in order to gain acceptance. If such constraints are present, establishments are advised to consider that every guideline implemented will improve the security of their systems.

Conversely, some environments and/or applications may demand a level of security significantly higher than the proposed baseline. This point was recognised within the research project and led to the definition of the protection methodology that will be described in chapter 5.

4.7 Conclusions

In conclusion, it is believed that the guidelines have fulfilled their objective of providing a solid basis for the improvement of security within existing HCE systems. Whilst baseline protection alone will not be sufficient in many scenarios, uniform adoption would provide the common foundation that the survey indicated was lacking.

With regard to further development of the baseline, it is envisaged that whilst the principles will remain relatively static, the underlying guidelines will require periodic updates to account for changes within the healthcare field or in the types of information system technology available (e.g. the increasing use of multimedia systems may introduce new
considerations). Changes within the local HCE (e.g. organisational structure, medical applications and practices) may also necessitate re-evaluation of some recommendations.

The involvement of the SEISMED reference centres has already ensured the general healthcare acceptability and applicability of the guidelines at a theoretical level. Further work is in progress at the time of writing in which selected guidelines are actually being implemented to provide a practical validation. These experiences will also be documented and available for future reference by European HCEs (AIM SEISMED 1995c).

Details of the guidelines are to be published in the proceedings of the SEISMED workshop “Security and Legal Aspects of Advanced Telematics Systems” (Brussels, 11 July 1994) and a separate paper has also been submitted to the international journal Medical Informatics. Both of these papers are included in Appendix F.

The next chapter will now proceed to consider how protection may be taken beyond the baseline level in the more sensitive healthcare scenarios.
CHAPTER 5

A Generic Methodology for Healthcare Data Security
5.1 Addressing healthcare security requirements above baseline

It is recognised that, whilst comprehensive, the baseline level recommendations will not be sufficient for all healthcare information systems and stronger protection will be required in some scenarios. The next question is, therefore, how these requirements may be best addressed.

Whilst the application of the baseline is relatively easy (insofar as all guidelines are generally meant to be considered in all systems), progressing beyond it will be more complicated in that requirements will be much more system or environment-specific. HCEs will need to determine where further protection is required and what level is appropriate. It is, therefore, necessary to have some method by which these factors can be determined.

5.1.1 Traditional risk analysis

It can be seen from chapter 3 that the large scale introduction of security in healthcare across Europe is hampered by the variety of different applications and system configurations that may be identified. The issue is further complicated by the variety of information that may be held and the fact that several different levels of data sensitivity may exist. These factors make it impossible to assert a single level of security that will be appropriate for all applications, without being excessive in some cases (which is why the guidelines discussed in chapter 4 could only recommend the minimum, baseline level of protection).
Chapter 5: A Generic Methodology for Healthcare Data Security

As a result, healthcare establishments require guidance on the selection of appropriate measures, as well as on where and how they should be incorporated into their systems. The commonly accepted means of achieving this is to conduct a risk analysis investigation, where a specialist IT analyst looks in detail at the value of systems and data and determines the specific threats and vulnerabilities that apply to the establishment involved. This then results in a tailored security package for the system. However, it is possible to identify two significant disadvantages:

1. a full risk analysis investigation can be a complicated and, therefore, time-consuming process involving the co-operation of many HCE staff, which may result in significant disruption of normal working activity;

2. the required specialists are unlikely to exist in most HCEs, which will necessitate the use of potentially costly outside consultancy.

These points are underlined by the observations of Gaunt (1992) in relation to the practical use of the CRAMM (CCTA Risk Analysis and Management Methodology) methodology in HCEs. Whilst CRAMM is the method currently advocated by the UK National Health Service (Barber et al. 1993b), the process of conducting a review was considered too involved and too difficult for healthcare middle management to undertake. The need for a “trimmed-down” approach was identified.
As a consequence of these points, it can be seen that such an analysis may be prohibitive in many HCEs. However, given that many of the threats and vulnerabilities in healthcare are not unique to individual establishments, a full analysis in each case may be largely unnecessary. It is, in fact, possible to provide a protection methodology in which the basic risk analysis has already been performed, allowing existing systems to be easily classified in order to determine their security requirements.

5.1.2 Requirements for a profiling methodology

By studying the care activities carried out by the hospitals, general practitioners and community health centres (as well as the various support services that are necessary to facilitate these activities) the framework of a generic protection methodology for the healthcare environment has been developed. This is intended to provide a simplified means by which HCEs may determine their own security requirements.

It was previously established in the discussion of the existing systems guidelines that security must be examined from the perspective of the whole system, considering all elements that may influence protection requirements. However, in providing a methodology there were two rather conflicting requirements:

- given that recommendations will need to be selected to suit the host system rather than being applied across the board, it is desirable to use a simpler structure than that provided by the existing systems security principles (so that the approach can still be easily understood and applied by non-specialists);
at the same time, the approach must not be so simplistic that it does not allow recommendations to be tailored (to some degree) to the host systems under consideration.

A compromise was, therefore, reached in which HCE systems may be categorised at a generic level, as shown in figure 5.1.

These elements have been incorporated into the framework of a methodology as shown in figure 5.2, illustrating the high-level steps involved in determining the security requirements of existing systems and selecting appropriate countermeasures. The rationale behind the methodology is that similar organisations and systems will have similar security requirements. From this came the concept of developing generic system protection profiles that could then be used in any establishment.

However, the number of possible system variations makes it impractical to derive a profile for each of them on an individual basis. The most workable approach was
considered to be the development of a series of smaller profiles targeting each key element of a system. It would then be possible, using appropriate combinations of the profiled elements, to generate high-level system profiles to account for the majority of healthcare IT scenarios. What the methodology therefore proposes is a “mix and match” approach to countermeasure selection.

![Diagram](image)

**Fig. 5.2: Existing Systems Protection Methodology Overview**

Although the purpose of the methodology is to remove the need for a specialist IT risk analyst, it would still need to be applied by someone with a high level of IT expertise in order that the significance of its recommendations can be properly appreciated. It is envisaged that HCE information systems administrators or security officers would suffice in this capacity.
As shown in figure 5.2, the methodology proceeds up to the point of final countermeasure selection. The subsequent implementation would then proceed in the same manner as suggested for the baseline recommendations in chapter 4.

5.2 Elements of the methodology

The main elements of the methodology will now be considered in more detail.

5.2.1 Computer Configurations

The computer configuration refers to any IT assets that are related to the information system under consideration.

At a high level it is possible to identify a relatively small number of elements that may be included in any given computer configuration, as illustrated by the breakdown in figure 5.3 below.

Fig. 5.3: Computer Configuration groups
Chapter 5: A Generic Methodology for Healthcare Data Security

It should be noted that whilst networks are included here they were primarily the responsibility of a separate SEISMED workpackage, which developed a specific protection methodology for them (Patel and Kantzavelou 1994). As such, attention to this aspect was minimised.

Protection countermeasures may be associated with each of these configuration types, under the general categories shown in table 5.1.

<table>
<thead>
<tr>
<th>Countermeasure Category</th>
<th>Example issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>Physical access; theft protection.</td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Maintenance contracts; alternative processing arrangements; backup procedures.</td>
</tr>
<tr>
<td>System</td>
<td>Authentication; logical access controls.</td>
</tr>
<tr>
<td>Procedural</td>
<td>Backup / recovery policy; policy for software usage; hardcopy control.</td>
</tr>
<tr>
<td>Personnel</td>
<td>Operational training; Information Technology awareness.</td>
</tr>
</tbody>
</table>

Table 5.1: Computer Configuration countermeasure groups

Individual systems would be analysed to determine which of these elements are present, and which countermeasures are therefore applicable.

5.2.2 Operational Environments

This considers the nature of the environment in which the IT assets of the establishment are actually located and used.
Table 5.2 indicates the main features of an environment that are considered to influence the level and type of security required. Appropriate combinations of these factors can be used to describe the majority of healthcare establishments from general practice (i.e. primary care) up to large hospitals.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Options</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Fixed / Mobile</td>
<td>A variable environment (e.g. as would be associated with a portable computer system) limits the environmental measures possible.</td>
</tr>
<tr>
<td></td>
<td>Rural / Urban / City</td>
<td>The geographical area in which the environment is located will provide a general indicator of the local population density, crime potential and likelihood of natural disasters.</td>
</tr>
<tr>
<td>Buildings</td>
<td>Single / Multiple</td>
<td>The number of buildings will determine access control and site security requirements.</td>
</tr>
<tr>
<td></td>
<td>Old / Modern</td>
<td>The age of a building may indicate the likely level of risk from fire, natural damage etc.</td>
</tr>
<tr>
<td>People</td>
<td>Number (low, medium, high)</td>
<td>The number and mixture of people within an establishment influences the requirements for access controls and personnel related measures.</td>
</tr>
<tr>
<td></td>
<td>Staff / Contract / Public</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2: Operational Environment categorisation

Again, the basic countermeasures appropriate to each type of environment can be identified and the key categories are outlined in table 5.3.

<table>
<thead>
<tr>
<th>Countermeasure Category</th>
<th>Example issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site Security</td>
<td>Building / site access controls; theft prevention measures.</td>
</tr>
<tr>
<td>Disaster Planning</td>
<td>Protection against fire, flood and natural disasters.</td>
</tr>
<tr>
<td>Procedural</td>
<td>Control of visitors; controls on smoking, eating and drinking.</td>
</tr>
<tr>
<td>Personnel</td>
<td>Job recruitment / termination; security awareness.</td>
</tr>
</tbody>
</table>

Table 5.3: Operational Environment countermeasure groups
5.2.3 Data Sensitivity

It should be evident from previous comments that data is the key aspect of healthcare information systems. As such, significant attention has been devoted to the issue of establishing a suitable means to assess data sensitivity within the methodology.

In many cases it will ultimately be the beliefs and circumstances of the data subject that determines the sensitivity of personal data. At the same time, HCPs might generally argue that all healthcare data should be considered equally sensitive. However, neither of these observations are of any real use in a practical sense for specifying protection and some means of general assessment is necessary. As a first step it is useful to identify that the sensitivity of data will be influenced by two main factors as shown in figure 5.4.

![Diagram showing factors of data sensitivity](image)

These aspects, along with a means of assigning sensitivity ratings, are discussed in the subsections that follow.
5.2.3.1 Overview of healthcare data requirements

A high level analysis reveals that the basic activities performed during care delivery fall into the sequence of operations shown in figure 5.5. It can be seen as an iterative process centred around the patient, incorporating various clinical and administrative services that may be called upon.

![Diagram of General care activity flow]

Fig. 5.5: General care activity flow

All stages of care delivery may produce or require various types of data - the type, sensitivity and quantity of which will depend upon the problems and requirements of the
patient involved. In addition, the various associated support services will also generate much information. It is likely that most (if not all) of this data will require protection to preserve its confidentiality, integrity and / or availability.

The different types of data may be categorised as shown in figure 5.6.

Operational data is related to day to day clinical care, involving information that is used to directly govern care decisions. This will generally have the greater requirements in terms of data integrity and availability. Further general subdivisions can be identified, this time based on the confidentiality demands of the information, as shown below:

- **General**

  Data relating to the vast majority of patients requiring healthcare services within an establishment.
• Special

Data that relates to patients who are also HCE staff members, or represent special categories within the community (e.g. VIPs, politicians etc.). Information disclosure is likely to have greater impact for these individuals and, as a consequence, there may be an increased likelihood of the data being the target of a breach (e.g. for blackmail or journalistic purposes).

• Sensitive

Data relating to patients with particularly sensitive health problems, such as AIDS or psychological disorders. The level of protection required in these cases will frequently be influenced by statutory control (e.g. UK Mental Health Act).

Non-operational data refers to information that does not directly influence patient care decisions. Such information may be related more towards planning and resourcing purposes (for example, analysis of trends and workloads), management and administration of non-clinical departments (e.g. finance, estates) or the control of supporting systems (e.g. air conditioning, telephone switchboards). Whilst not directly associated with care activity, this data is still essential to the optimal functioning of the HCE.

Whilst these categories are useful as a general guideline, it is necessary to have a more detailed breakdown of medical data before it becomes possible to assign suitable protection with any accuracy. The approach taken to accomplish this was to develop a generic model of all major data types used in healthcare.
5.2.3.2 A Healthcare Generic Data Model

The decision was taken to develop a generic data model to provide a simple means of specifying what data is available within a system and thereby help with this stage of allocating protection.

A number of existing healthcare data models were considered (including the Korner Data Model (NHS 1985) and the NHS Common Basic Specification (NHS 1990)) but none appeared to be viable for use in security specification. The main problems were that the models were either too detailed to be practically useful or had not been devised with security considerations in mind. A simplified model of the information stored within the HCEs was therefore required.

At a very broad level, the range of sensitivity for healthcare-related data can be defined as follows (Commission of European Communities 1991b):

\[
\text{name} \rightarrow \text{Administrative} \rightarrow \text{Personal} \rightarrow \text{Highly sensitive personal data} \rightarrow \text{data} \rightarrow \text{data} \rightarrow \text{data} \rightarrow \text{data} \rightarrow \text{& diagnostic data}
\]

However, this breakdown is rather too simplistic for security specification purposes, in that each of the last three categories can be seen to encompass a very wide range of information. As a result, varying levels of sensitivity will exist within them.

In order to establish a more comprehensive high-level view of data requirements, a number of basic system arrangements were examined from various establishments within Europe (including the SEISMED Reference Centres and further hospitals located in Hannover,
The key groups of data utilised by these systems were then identified, and structured into an initial model which was considered to be an appropriate foundation for specifying protection guidelines. Subsequent refinements were then made based on opinions gathered from various healthcare personnel.

The new model is comprised of twelve data groups as shown in figure 5.7, providing a framework that should encompass all data required by a HCE.

Fig. 5.7: Healthcare Generic Data Model
Chapter 5: A Generic Methodology for Healthcare Data Security

The diagrammatic form shown in the figure is intended to provide a consistent and conceptually simple means of indicating and representing the data utilised by different healthcare applications. It should be noted that the groups specified can also adequately include all of the healthcare data types that have previously been cited as being important from the security viewpoint (Gritzalis et al. 1991).

The data groups identified by the model are of a high-level nature, but they can be subdivided into further levels of detail as required. The following pages include a breakdown of each data group, including a brief description of the type of information held and example data items.

1. Patient Identification

This group describes the general information that is held regarding each patient referred to the healthcare service. It is often used in a number of systems within the organisation. The data is split into demographic and social subcategories. Demographic data is largely for identification purposes, whereas social data may also be required in order to determine additional risks, genetic predisposition and the need for modifications to normal care. The disclosure of social data generally has the greater potential for embarrassment.

Example data:

Demographic

Name
Chapter 5: A Generic Methodology for Healthcare Data Security

Address with postcode
Age,
Sex,
Ethnic origin
Marital status,
Next of Kin,
Identifiers (e.g. National Insurance or social security numbers).

Social
Religion,
Occupation,
Sexual orientation.

2. Patient Administration

Data that describes patients contact with the HCE. This group contains information used in the day to day scheduling of the various non-clinical care activities carried out on patients (e.g. the delivery of resources that in turn facilitate clinical care). It allows for the planning and efficient running of a large number and complex sequence of system operations.

Example data:

Waiting List

Time and date of Appointments
Name of clinic
In patient or Out patient
Chapter 5: A Generic Methodology for Healthcare Data Security

Location
Consultant

Theatre Management
Scheduling of operations
Consultants
Anaesthetists
Routine or emergency
Inventory

Ward and Bed Management
Patient Groups
Specialities
Day care details
Service Capacity

Transport
Emergency and routine allocation
Routes
Times and dates

Referral Details
Method, Source
Date of admission
Intended stay
Consultant

Discharge Details
3. Patient Care

This group contains the medical history, diagnosis, care decisions and the treatment information relating to individual patients. It is essentially the electronic representation of the patient healthcare record.

Example data:

- Episode Information
- Dates of admissions / discharges
- Staff Involved
- Diagnosis including Clinic Coding/s
- Care Plan
- Specific needs
- Healthcare Delivered
- Drug Therapy
- Outcome of treatment
- Consultants and anaesthetists reports

4. Clinical Services

This data group is obtained from the analysis facilities within the HCE. It is concerned with the operational functioning of service departments and the data generated is for internal use within the departments (i.e. not patient related).
Example data:

Radiology Information
Pharmacy Information
Laboratory Information

Diagnostic Requests
Type
Source
Work Order
Specimen details
Investigation details
Intermediate Results

Pharmacy
Prescription
Dispensing details
Product Issue
Stock control details

5. Finance

This group of data covers all aspects of finance involved in the operation of healthcare organisations. Data here is used for payroll, acquisition and economic modelling purposes. Such information could conceivably be of use to competing HCEs and could disrupt operations (including the
potential to indirectly affect patient care activity) if damaged or unavailable for long periods.

Example data:

- Contracts
- Payroll
- Invoicing
- Purchasing
- Budgets

6. Hotel Services

This group covers the information stored on all of the basic "housekeeping" functions within healthcare establishments. Principal examples include catering, portering, transport and cleaning services.

Example data:

- Hospital supplies
- Catering
- Domestic / cleaning
- Works data

7. Staff

This group includes personnel information on all grades of staff working within the HCE.
Example data:

Personnel details

Rostering information


This group involves the data used in management, monitoring and planning of healthcare organisations.

Example data:

Statistics on planned intention, unmet demand

Ward occupancy

Clinic activities

Emergency

Day care

Transport

Planning and estimating

General management data

9. Library and Information Services

This data group encompasses the existing medical knowledge that is referenced by clinical staff and national / local protocols for clinical management. Such data is not normally linked to individual patient
records (although it may be referenced in order to justify a clinical decision).

Example data:

Medical Knowledge

Drug Information

Definitions - Codings

- Classifications

Care guidelines, procedures and protocols

10. Expert Systems

This group represents information utilised by decision support tools and / or neural networks within the HCE. These may directly assist in the planning of healthcare to determine the most appropriate care for individual patients, or in the overall running of the HCE. Data integrity is obviously highly important for the correct / effective operation of such tools. This data is represented as a separate group as it also refers to the data that is used to control the expert system as opposed to just the information that is presented to users. In addition, systems offer the power of deduction and, as such, the accessible information will not necessarily be predefined, making sensitivity more difficult to quantify.

Example data:

Expert knowledge
Chapter 5: A Generic Methodology for Healthcare Data Security

11. Communications Services

This group serves to identify the process of internal communication within the HCE and takes into consideration the different formats by which data is transmitted.

The majority of communicated information would ultimately reside in another data group of the model. The nature of shared healthcare information systems essentially means that the storage of data by one individual will ultimately lead to its communication to other staff. Conversely, communication is often the starting point for a large amount of patient data (for example, information on all services required and conducted on behalf of the patient) and may, therefore, be potentially sensitive. Additionally, it may be possible either to determine directly or infer information on service levels or HCE productivity (e.g. from communications such as staff work planning, supply orders, requisition of tests and return of results), all of which requires protection.

The group could contain a wide range of additional data that is generated during the communication of information around the organisation (i.e. e-mail, transaction information, requests for activity, general notices).
Example data:

- Messages
- Work orders
- Forms
- Results
- Referral notes
- Purchase requisitions

12. External Systems

This group recognises the potential for substantial data relationships / flows to exist between different applications. The incorporation of this group into a mapping may therefore be used to represent the "interface" to another system.

A specific application may store or communicate data from all of these groups, or a particular subset of them. It is consequently possible to map real world systems onto the model, indicating the data groups that are involved. This is effectively the first step in establishing the data sensitivity of the system. Figure 5.8 shows how the mapping of a Patient Administration System (as found in the Plymouth Reference Centre) could appear.
Further example mappings, based upon operational systems found within Plymouth and Thessaloniki hospitals can be found in appendix C. These were specified by the establishments concerned, who were amongst those given access to the model and data group descriptions for evaluation purposes. The resulting mappings should, therefore, give an accurate representation of each systems data requirements and further show how the
model can be applied in practical scenarios. The applications covered include radiology and mental health systems (Plymouth) and staffing, accounting, pharmacy and patient administration systems (Thessaloniki). This last example can be usefully contrasted with the system represented in figure 5.8, in that whilst they represent the same basic application, different data groups are encompassed. This in turn indicates that an alternative approach to protection profiling that was considered (based upon specifying standard profiles for generic types of healthcare application, such as ‘Patient Administration System’) would have been impractical.

The generic data model was the subject of a paper presented at the MIE 93 Medical Informatics Congress in Jerusalem during April 1993 (Sanders and Furnell 1993), a copy of which can be found in Appendix F.

5.2.3.3 Data Use

Whilst the model allows easier identification of the types of data present in a system, it has already been stated that data sensitivity is also affected by the context in which the information is used. As such, it is also necessary for the methodology to incorporate a similarly high level set of data uses.

Related work within the SEISMED project (AIM SEISMED 1993a) identified a generic set of nine healthcare data uses that were suitable for the model. However, this work only provided a simple description of the different uses and did not attempt to relate them to any corresponding requirements for security. The nine categories are described in table 5.4, with further information relating to sensitivity rating being added in the next section.
Chapter 5: A Generic Methodology for Healthcare Data Security

<table>
<thead>
<tr>
<th>Data Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Clinical</td>
<td>Used in the planning, delivery and monitoring of patient healthcare.</td>
</tr>
<tr>
<td>Emergency Care</td>
<td>Provision of care in a clinical emergency, where optimal conditions and / or information cannot be guaranteed. Therefore, only a minimum set of essential data is required, with HCPs relying on their own training and experience.</td>
</tr>
<tr>
<td>Critical Clinical</td>
<td>Control of instrumentation / systems in direct feedback loops (e.g. control of radiation dose administration to cancer patients). Data availability and integrity essential in such contexts.</td>
</tr>
<tr>
<td>Expert Systems</td>
<td>Use in decision support tools or neural networks, which aid clinical diagnosis and interpretation or general management of HCE.</td>
</tr>
<tr>
<td>Operational Non-clinical</td>
<td>Use of information that supports the HCE infrastructure, but does not directly influence the care of individuals.</td>
</tr>
<tr>
<td>Financial</td>
<td>Use of data in financial systems for contract management, purchasing and patient billing.</td>
</tr>
<tr>
<td>Planning &amp; resource management</td>
<td>Systems used for aggregation of patient data for planning and clinical review purposes.</td>
</tr>
<tr>
<td>Clinical Research</td>
<td>Identifiable or anonymised data used for research purposes. Normally utilises aggregated data.</td>
</tr>
</tbody>
</table>

Table 5.4: General categories of medical data usage

5.2.3.4 Approach to sensitivity rating

Data sensitivity has been considered in terms of the following key impacts that may result from lack of protection, covering the issues of confidentiality, integrity and availability:

- information disclosure;
- denial of access to data;
- modification of data;
- destruction of data.
These impacts are rated at different levels, with different sets of countermeasures being associated with each level of each impact type. For simplicity, ratings are low, medium or high (where low represents basic protection and high is the maximum level of countermeasure available). The impact level has been assigned by considering a number of potential consequences (as previously identified in chapter 2):

- loss of confidentiality;
- disruption of activities;
- embarrassment;
- failure to meet legal obligations;
- financial loss;
- threat to personal safety.

The potential impacts relate to the types of data involved and the way(s) in which it is used. The impact from disclosure is most closely related to the data type involved. Data will generally portray the same information in all contexts and the protection afforded should therefore remain consistent regardless of the application that uses it. Conversely, any impact resulting from denial, modification or destruction of data is heavily influenced by the purpose for which it is being used.

As an example we may consider patient information indicating that an individual is a registered drug addict. When used in the context of direct care the denial, modification or destruction of such information could adversely affect care delivery and hence the
requirement for protection would be quite high. However, if the same information was used in a less critical context (e.g. generation of a statistical summary of addict cases), then the resulting impacts could be somewhat less. In either case, the raw data would be the same, and hence the impact from disclosure would remain the same.

A set of general impact ratings are presented in tables 5.5 and 5.6. These valuations were determined using the following strategy. In the case of individual data types and uses, the rating levels were influenced by considering the six factors above. As an example of this, table 5.7 lists the main factors that influenced the ratings for the data type Patient Care and the Operational Clinical data use. At a higher level, more general criteria were considered, such as the need to maintain the integrity of patient care data. From this, it can be seen that the data modification impact is rated high in all contexts that may relate to care delivery.

Finally, opinions were gathered from various European medical practitioners, using a small survey that contained the set of initial valuations and asked for feedback on their validity (AIM SEISMED 1993c). This served to provide some level of validation and the final values are based upon an amalgamation of the responses received (and should, therefore, represent a reasonable view of the issue).

Having first identified the data types and uses in a system, it is then possible to use these ratings to determine the appropriate levels of protection countermeasure. Where a number of types and uses are identified, the extraction of the highest impact values will allow protection to be delivered that caters for the worst case scenarios.
### Table 5.5: Sensitivity ratings for data disclosure

<table>
<thead>
<tr>
<th>Data Group</th>
<th>Disclosure Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Identification</td>
<td>Low</td>
</tr>
<tr>
<td>Patient Administration</td>
<td>Medium</td>
</tr>
<tr>
<td>Patient Care</td>
<td>High</td>
</tr>
<tr>
<td>Clinical Services</td>
<td>Low</td>
</tr>
<tr>
<td>Finance</td>
<td>Medium</td>
</tr>
<tr>
<td>Hotel Services</td>
<td>Low</td>
</tr>
<tr>
<td>Staff</td>
<td>Low</td>
</tr>
<tr>
<td>Resource Management &amp; Planning</td>
<td>Medium</td>
</tr>
<tr>
<td>Library &amp; Information Services</td>
<td>High</td>
</tr>
<tr>
<td>Expert Systems</td>
<td>Medium</td>
</tr>
<tr>
<td>Communications Services</td>
<td>High</td>
</tr>
<tr>
<td>External Systems</td>
<td>High</td>
</tr>
</tbody>
</table>

### Table 5.6: Sensitivity ratings for data denial, modification & destruction

<table>
<thead>
<tr>
<th>Data Use</th>
<th>Denial Impact</th>
<th>Modification Impact</th>
<th>Destruction Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Clinical</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>Emergency Care</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>Critical Clinical</td>
<td>High</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Expert Systems</td>
<td>Low</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Operational non-clinical</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Financial</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Planning &amp; Resource Mgmt</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Quality Management</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Clinical Research</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>

### Table 5.7: Derivation of sensitivity ratings

<table>
<thead>
<tr>
<th>Data Type or Use</th>
<th>Impact Type</th>
<th>Rating</th>
<th>Reason / Derived from</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Care</td>
<td>Disclosure</td>
<td>High</td>
<td>confidentiality, embarrassment, legal</td>
</tr>
<tr>
<td>Operational Clinical</td>
<td>Denial</td>
<td>Medium</td>
<td>disruption, safety</td>
</tr>
<tr>
<td></td>
<td>Modification</td>
<td>High</td>
<td>safety, legal</td>
</tr>
<tr>
<td></td>
<td>Destruction</td>
<td>Medium</td>
<td>disruption, safety</td>
</tr>
</tbody>
</table>
Whilst these valuations serve to give an illustration of the concept, it would be possible to specify them in greater detail by using a more extensive breakdown of impact types and levels. However, this was not considered necessary during the development of the methodology (see section 5.6.1 for further discussion of this aspect).

5.3 Countermeasure selection

Security countermeasures can be identified and refined at various stages within the methodology. The overview diagram from Figure 5.2 categorised them under three headings, which are distinguished as follows:

1. Basic system countermeasures

These represent the minimal security considerations for a given type of computer configuration operating in a particular environment, and should be considered irrespective of the data held or the purpose(s) for which the system is used.

2. Appropriate countermeasures

These represent the overall set of countermeasures that may be appropriate for a given system, having also considered what data is used and how. This does not take into account any practical constraints that may apply in respect to implementation.

3. Selected countermeasures

Whilst the "appropriate countermeasures" may represent an ideal solution, a number of real world factors are also likely to influence the final selection process (identified as...
"other factors" in figure 5.2). These are principally considered to include the following elements:

- **Cost constraints**
  The cost of adopting particular countermeasures may be considered from several perspectives (e.g. financial, performance, practicality etc.). Acceptable levels will obviously be highly dependent upon individual environments and their priorities. As previously identified, financial cost is perceived to be a particularly important factor in security-related decision making for the majority of healthcare establishments.

- **Operational constraints**
  The nature of the organisation itself will also influence countermeasure selection. Proposals should not conflict too greatly with the established practice of the particular healthcare environment, or they risk being rejected. This relates to the idea of "staff culture" that was previously identified in section 2.4.4.

- **Existing countermeasures**
  Any security countermeasures that are already in place will obviously influence whether some of the suggested countermeasures need to be considered / adopted.

These are obviously subjective elements in the application of the methodology and, as such, it is not possible to formalise them further.
The "selected countermeasures" represent the final output of the methodology (having considered any limitations of the individual HCE), which may be added to the existing system to address its security requirements.

Given that the objective of the work was to establish a methodology framework, the definition of a full range of accompanying countermeasures was outside the scope of the research programme. However, the issue is being addressed as part of a further research project which is currently building upon the foundation that has been provided here (Warren 1995). For completeness, illustrative examples of the types of countermeasures that might be recommended are included in the example scenario in section 5.5.

5.4 Formal stages of the methodology

This section describes the formal steps by which the methodology may be implemented.

Countermeasures are derived as shown in figure 5.9. Appropriate countermeasures would be selected at each stage from corresponding categories (note that some duplication would be
likely to occur in terms of the suggestions arising). The stages of the methodology may be more formally described as shown in table 5.8.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Determine basic system profile</td>
</tr>
<tr>
<td>Inputs</td>
<td>None</td>
</tr>
<tr>
<td>Output</td>
<td>Basic system countermeasures</td>
</tr>
<tr>
<td>Description</td>
<td>Categorisation of the computer configuration and operational environment of the existing system according to pre-determined profile categories. For computer configuration choose appropriate elements from: Laptop / Portable, Desktop PC, Mini / Mainframe, Network. For operational environment categorise elements of: Location, Buildings, People.</td>
</tr>
</tbody>
</table>

| 2     | Determine data sensitivity |
| Inputs | None |
| Output | Data-related countermeasures |
| Description | Establishment of data types and uses. Selection of countermeasures based upon sensitivities encompassed. Choose countermeasures from each of the following groups based upon sensitivity levels identified: Disclosure countermeasures, Denial / Destruction countermeasures, Modification countermeasures. This stage is described further in section 5.4.1. |

| 3     | Determine appropriate system countermeasures |
| Inputs | Basic system countermeasures; Data-related countermeasures |
| Output | Appropriate system countermeasures |
| Description | Combination of the countermeasures obtained from stages 1 & 2 to form an initial countermeasure set that would satisfy the requirements of the existing system. |

| 4     | Select final countermeasures |
| Inputs | Appropriate countermeasures |
| Output | Selected (final) system countermeasures. |
| Description | Refinement of the countermeasure set by considering any HCE specific factors / constraints that may apply. |

Table 5.8: Formal stages of the protection methodology
5.4.1 Determining Data Sensitivity

Determining the countermeasures dictated by data sensitivity is the most complex stage of the methodology, as they will be based upon a variety of impact values derived from the data involved. It is necessary to establish:

- impact valuations for disclosure (based on data type only);
- impact valuations for denial, modification, destruction (based on data uses).

The specific procedure involved is described in table 5.9 and illustrated in figure 5.10.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Identify the data groups involved (using the generic data model).</td>
</tr>
<tr>
<td>2.2</td>
<td>Determine disclosure impacts from the model group valuations.</td>
</tr>
<tr>
<td>2.3</td>
<td>Identify general data usage category(s) that apply to the system.</td>
</tr>
<tr>
<td>2.4</td>
<td>Determine denial, modification and destruction impacts from usage valuations.</td>
</tr>
<tr>
<td>2.5</td>
<td>Derive overall sensitivity values for the application by selecting &quot;worst case&quot; values from component groups (i.e. 4 values in total). From this the appropriate data sensitivity countermeasures may be selected.</td>
</tr>
</tbody>
</table>

Table 5.9: Stages of data sensitivity assessment
5.5 An example of methodology implementation

The following section presents a basic example to illustrate the application of the methodology. It is based upon a typical information system scenario that may be found
within the UK health service (namely the Patient Administration System example introduced earlier in this chapter). It has not been possible to apply the methodology in practice at this stage in time and the example is, therefore, a theoretical outline of how the procedure would work.

As previously mentioned, the countermeasures shown would be selected from predetermined lists. However, the example provides only a small, representative selection of what might be recommended.

It should also be noted that the example only proceeds to the third stage of the methodology. The reason for this is that stage 4 relates to the consideration of subjective factors in specific real world environments. It was felt that the imposition of artificial constraints would add little to the example.

**Scenario Outline**

A patient records system is maintained by a small primary care practice. The system is primarily based upon a standalone PC, although selected data may be transferred to and from this using a portable computer that the healthcare practitioner takes on general visits and emergency call-outs. The practice is based in a single, modern building located in an inner city.

**Stage 1 : Determine basic system profile**

The following factors can be determined from the scenario description:
Chapter 5: A Generic Methodology for Healthcare Data Security

Computer Configuration: Laptop / Portable - Standalone;

Desktop PC - Standalone;

Operational Environment: Location - Fixed & Mobile, City;

Building - Single, Modern;

People - Staff, Public, Low.

Stage 2: Determine Data Sensitivity

2.1 Identify data groups

Three data groups are encompassed (and can be identified using the data model as previously shown in figure 5.8), these being:

- Patient Identification;
- Patient Administration;
- Patient Care.

2.2 Determine disclosure impacts

The following ratings can be extracted from the disclosure impact valuations previously given in table 5.5:

<table>
<thead>
<tr>
<th>Data Group</th>
<th>Impact Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Identification</td>
<td>Low</td>
</tr>
<tr>
<td>Patient Administration</td>
<td>Medium</td>
</tr>
<tr>
<td>Patient Care</td>
<td>High</td>
</tr>
</tbody>
</table>
Chapter 5: A Generic Methodology for Healthcare Data Security

2.3 Identify data uses

Potential uses of the data in a Patient Administration System are determined as being:

- Operational Clinical;
- Emergency Care.

2.4 Determine Denial, Modification & Destruction Impacts

The following impact valuations can be extracted from table 5.6 relating to data denial, modification and destruction.

<table>
<thead>
<tr>
<th>Data Use</th>
<th>Denial Impact</th>
<th>Modification Impact</th>
<th>Destruction Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Clinical</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>Emergency Care</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
</tbody>
</table>

2.5 Derive overall sensitivity ratings

The "worst case" values from the previous tables are used to determine the overall sensitivity of the system.

- Disclosure : High
- Denial : Medium
- Modification : High
- Destruction : Medium

These are now used to determine the level of data sensitivity countermeasures.
Stage 3: Determine appropriate system countermeasures

The overall countermeasure sets that would be considered in this scenario are as follows.

Computer Configuration Countermeasures

<table>
<thead>
<tr>
<th>Countermeasure Category</th>
<th>Example countermeasures</th>
<th>Laptop / Portable (Standalone)</th>
<th>Desktop PC (Standalone)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>Casing locks</td>
<td>Locks and / or alarms</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Property markings (visible and UV)</td>
<td>Property markings (visible and UV)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Protective carry case</td>
<td>Site to deny casual access</td>
<td></td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Service warranty</td>
<td>On-site service contract</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Maintain / store data backups</td>
<td>Maintain / store data backups</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Carry spare batteries etc</td>
<td>Documented / tested recovery strategy</td>
<td></td>
</tr>
<tr>
<td>System</td>
<td>Use of any standard security features</td>
<td>Use of any standard security features</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Password protection</td>
<td>Password protection</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Virus checking</td>
<td>Virus checking</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hard disk encryption</td>
<td>Menu-only access (no DOS)</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>Store sensitive data on separate media</td>
<td>Use of any standard security features</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Care of floppy disks</td>
<td>Integrity checksums</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Lock away when not in use</td>
<td>Ban unauthorised software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Regular backup to desktop machine</td>
<td>Control software updates</td>
<td></td>
</tr>
<tr>
<td>Personnel</td>
<td>Stress individual accountability for machine / data when offsite</td>
<td>Provide software training</td>
<td></td>
</tr>
</tbody>
</table>

Operational Environment Countermeasures

Single-building / Modern / City

<table>
<thead>
<tr>
<th>Countermeasure Category</th>
<th>Example countermeasures</th>
<th>Mobile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>Use of staff ID badges</td>
<td>The nature of this environment is, by definition, variable, making it difficult to cite environment-specific countermeasures.</td>
</tr>
<tr>
<td></td>
<td>Receptionist / guard at main entrance</td>
<td>Additional attention should, therefore, be devoted to the physical countermeasures relating to the computer configuration, with the level of protection being appropriate to account for the &quot;worst case&quot; scenario.</td>
</tr>
<tr>
<td></td>
<td>Room access control (locks)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Alarm systems</td>
<td></td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Smoke and moisture detectors</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fire alarm (linked to fire station)</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>Visitors escorted (non-public areas)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Strangers challenged (non-public areas)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prohibit smoking</td>
<td></td>
</tr>
<tr>
<td>Personnel</td>
<td>Controlled access hours</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Defined responsibilities</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Monitor maintenance work</td>
<td></td>
</tr>
</tbody>
</table>

111
Data Sensitivity Countermeasures

**Disclosure**

<table>
<thead>
<tr>
<th>Countermeasure Level</th>
<th>Example countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>File-level passwords</td>
</tr>
<tr>
<td></td>
<td>SMART cards</td>
</tr>
<tr>
<td></td>
<td>Hard copy controls</td>
</tr>
<tr>
<td></td>
<td>Encrypted transmission</td>
</tr>
<tr>
<td></td>
<td>Encrypted storage</td>
</tr>
<tr>
<td></td>
<td>Removable storage media</td>
</tr>
<tr>
<td></td>
<td>Secure disposal of media / paper</td>
</tr>
<tr>
<td></td>
<td>TEMPEST protection</td>
</tr>
</tbody>
</table>

**Modification**

<table>
<thead>
<tr>
<th>Countermeasure Level</th>
<th>Example countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>File-level passwords</td>
</tr>
<tr>
<td></td>
<td>Integrity checksums</td>
</tr>
<tr>
<td></td>
<td>Auditing</td>
</tr>
<tr>
<td></td>
<td>Digital Signature</td>
</tr>
<tr>
<td></td>
<td>Data Encryption</td>
</tr>
</tbody>
</table>

**Denial / Destruction**

<table>
<thead>
<tr>
<th>Countermeasure Level</th>
<th>Example countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>Regular recovery checks</td>
</tr>
<tr>
<td></td>
<td>Alternative processing arrangements</td>
</tr>
<tr>
<td></td>
<td>Disk shadowing</td>
</tr>
<tr>
<td></td>
<td>Resource control</td>
</tr>
</tbody>
</table>

5.6 Extension of the methodology

Whilst the methodology is considered to provide a reasonably comprehensive means of classifying the additional protection requirements of existing systems, there are several ways in which it could be extended or improved. Two principal ideas are presented in this section.

5.6.1 Enhanced system classification

There is definite potential to extend the level of classification for each of the key information system elements used in the methodology. This has the obvious advantage that allowing greater levels of specificity in profiling will result in the countermeasures being increasingly tailored to the needs of the host systems.
With regard to the computer configuration, further subdivision of the main groups could be performed according to factors such as:

- number of users;
- number of terminals / potential access points;
- modes of access available (e.g. local or remote);
- number of applications supported;
- operating system security provision.

Classification of operational environments could be assisted by providing more specific predefined profiles. For example, typical countermeasure sets could be determined to suit:

- primary care establishments;
- community hospitals;
- general hospitals;
- specific medical departments.

Whilst the data model and data uses are considered to provide a solid high-level breakdown, it is conceivable that in some cases a more specific division of the existing groups could be advantageous. For example, the existing Patient Care data group could be subdivided into further the groups Psychiatric, Diagnostic and Treatment as suggested in related work by Davey (1994).
The way in which data sensitivity is assessed has considerable scope to allow more specific ratings. For example, it would be possible to utilise more detailed impact scales instead of the current low-medium-high ranking. This could include the possibility of rating the separate contributors to the overall impact (i.e. impact in terms of patient safety, embarrassment, financial loss etc.). This was, in fact, the approach used during initial development of the data model aspect, with each of the impact factors being assessed using a 1-10 ratings scale (as used in CRAMM). However, this was felt to be far too complex for what was intended as a simplified methodology and it was also considered difficult to separate countermeasure recommendations to this degree.

The rating of data uses could also be extended if required, for example by rating each data type in each usage context (although again this could serve to make matters significantly more complicated).

All such changes would serve to complicate the process of applying the methodology and would, therefore, place further restrictions upon who would be capable of doing so. In addition, of course, the more detail that is added, the closer the methodology will come to representing a full scale risk analysis in terms of the time and effort required to apply it (remembering that these points were the deficiencies that the approach was originally intended to overcome).

It is envisaged that further complexity could only be introduced effectively if the methodology was also to be enhanced with expert system technology, as suggested in the next section.
5.6.2 Expert system implementation

During the course of the methodology development the potential was identified for implementing the idea within an expert system framework. Although the full development of such a system was considered outside the scope of the research, a conceptual design was undertaken for discussion purposes and is described in this section.

The main purpose of the system would be to provide an intelligent decision support tool to assist in applying the methodology, based on standard expert system techniques (Giarratano and Riley 1989). It is considered that the most appropriate approach would be for the system to be based around a "consultation" style of interaction, guiding the user through each stage of the security analysis process.

The expert system knowledge base would contain a full range of countermeasures and selection rules associated with the methodology, along with additional expertise gathered from security consultants to enable further inferences (the latter would be related more to selection rules than actual countermeasures, being based upon the experts own experiences).

The principal stages of the expert system analysis would correspond closely to the normal steps in applying the methodology, with the system eliciting a fundamental system description from the user (identifying the computer configuration, operational environment(s) and data groups / uses involved). The majority of the user interaction would occur at this stage, with the system querying the user to establish which elements are present. The level of expertise employed could be made dependent upon the security and
IT experience of the user. Therefore, in the case of the analysis being driven by a relatively novice user, the system would rely upon a detailed style of consultation in order to elicit the required knowledge. Conversely, experienced staff would be more likely to utilise the system as an automated methodology tool.

An initial system profile would be derived from the consultation using a series of basic selection rules associated with each methodology category. At this stage countermeasures could be extracted directly from the knowledge base without any need for further inference.

The basic profile would not take into account any of the practical constraints that might apply with regard to countermeasure implementation (i.e. the financial limitations, operational constraints and / or existing countermeasures). Further consultation to establish such constraints could be used as the basis for filtering of the countermeasure suggestions (helping to automate the transition from appropriate to selected countermeasures). This would, however, demand that the data in the knowledge contained information about both “implementation difficulty” and costs (the latter of which would need to be updated regularly in order to be practical).

Having established the basic profile and any constraints, more advanced selection rules could be utilised to allow inferences based upon information from across several categories (which would be based upon the additional knowledge gathered from the experts). This would potentially allow the identification of additional requirements that may have been missed during the initial consultation.
Chapter 5: A Generic Methodology for Healthcare Data Security

It is anticipated that the overall structure of the system, and the process of user interaction involved, would be as illustrated in figure 5.11.

Fig. 5.11: Expert System Structure and Interaction

It is envisaged that the use of expert system technology would provide a number of advantages. At the most basic level it would serve to ensure the correct and consistent application of the methodology concepts. However, the encapsulation within this framework would also offer opportunities beyond the simple automation of the methodology. Having established the basic system profile by following through the key methodology stages, the specification could then be enhanced using inferences based upon the advanced rules in the knowledge base. The countermeasure recommendations would then be narrowed, making them more specific to the system under consideration. Finally, an
expert system would improve the user friendliness and general accessibility of the method. It would improve the opportunity for the techniques to be employed by healthcare staff who were not necessarily fully security-trained (e.g. the hospital IT manager). If such a system were to be developed for the PC environment then this would guarantee the maximum potential for adoption, given that this platform is available in nearly all HCE environments.

Although the expert system approach was not pursued beyond the conceptual stage within this project, an actual implementation is being undertaken within the related research programme that was previously mentioned in section 5.3.

This conceptual design was presented at NNESMED 94, an international conference on the use of neural networks and expert systems in healthcare and medicine, which was held in Plymouth in August 1994. A copy of the paper appears in appendix F.

5.7 Conclusions

The methodology serves to illustrate how high level categorisations of healthcare systems may be used to considerably simplify the process of security selection. It is envisaged that the approach would be valuable in cases where a full security review has previously been denied on the grounds of budget or inconvenience.

A fully developed methodology of this type should be usable with the majority of scenarios, catering for a range of general existing system categorisations. Despite this, however, it is still conceivable that some systems would be encountered that do not fit comfortably within the profiles suggested. In these cases it would still be necessary to perform a more detailed
risk analysis to determine the specific requirements of the system / environment. In any case, the recommendations from the methodology would always need careful, common-sense consideration, especially if they appear either too low or too high for the system involved. Again, specialist advice would be advocated in cases of significant doubt or where extremely high levels of risk are identified.

The methodology as described has been published in *Medical Informatics* (Furnell et al. 1994) and a copy of the paper appears in appendix F. In addition, although it was not originally one of the specified deliverables, the approach was also the subject of a supplementary report submitted to the CEC as part of the work from the SEISMED project.
CHAPTER 6

Improving system security in healthcare
6.1 Introduction

It should be evident that the security strategies advocated in chapters four and five were intended to address healthcare requirements as a whole, encompassing all of the key areas in which protection may be required. As a result, the approach presented so far has remained necessarily broad and largely theoretical.

However, the scope of the research programme also allowed for the investigation of more practical, technically-based means of improving existing systems security. As such, the focus of the report now changes significantly to examine a specific category of technical measures that are considered appropriate for use in healthcare.

With regard to the work presented so far, it can be seen that whilst the overall approach to security classification and enhancement is new, the underlying recommendations made by the guidelines and the protection methodology are largely confined to those which can be readily accomplished using existing technologies. This reliance was necessary, given that the techniques needed to be usable now by European HCEs. However, there are a number of areas in which considerably more advanced security techniques could be applied to existing healthcare systems and an examination of these will be the principal focus of the remainder of the thesis.

It was previously established in section 3.2.4 that existing security measures are often weak in healthcare systems, leading to (amongst other things) reports of the following undesirable incidents:


- user abuse of systems;
- internal hacking;
- external hacking;
- viruses.

Although the earlier results indicated that such malicious activities are normally more infrequent than other types of security incident, they are potentially the most costly to the organisation (with possible impacts ranging from the simple disruption of HCE activity to threatening patient safety). The key to combating and preventing them lies in improved logical security measures on the system side.

The conventional approach to logical security (as highlighted in the Authentication and Access Control principle of the new guidelines) is to use suitable techniques to create a "shield" around the system, with a consequent reliance that it cannot be penetrated. However, there are a number of limitations to this, particularly in the context of healthcare and existing systems, as identified below.

- The approach may constrain the user in many circumstances, introducing additional barriers that may be inconvenient to legitimate users. This is obviously undesirable in the healthcare environment.

- In some cases addition / enhancement of controls may not be straightforward and adoption may necessitate significant changes to, or even abandonment of, existing systems. Even if the approach was desired, such change could only be phased in
over a relatively long period. In healthcare it would be likely to be infeasible on grounds of cost.

- The approach tends to rely on the creation of an absolutely secure system. This may be unrealistic for several reasons, including flaws in both system design and subsequent administration.

- Finally, the resulting system may still be vulnerable to abuse by authorised users who misuse their rights. Numerous sources (American Bar Association 1984; Evans 1991) have indicated that as many as 80% of security incidents are the result of an organisation's own staff.

These limitations lead to the requirement for a different approach to system security which is not totally reliant upon preventing intruders from gaining initial access.

Authentication in existing systems is often solely based upon the use of passwords. The probable reasons for this are the convenience to the user and the expense that is often incurred by more elaborate techniques. In healthcare the issues of authentication and access control are complicated by the fact that many areas of an establishment will be open to the public. Terminals will often be widely distributed, with a necessity for many in public areas (e.g. on the wards). As a result it is not always possible to implement sufficient physical security or to rely upon continuous manual supervision.
However, in a domain such as healthcare, it would be foolish to insist upon a level of security that would greatly impede users in their legitimate work (i.e. security at the expense of care delivery). This points to a requirement for a security system that can operate transparently unless abuse of some kind is suspected.

A further consideration is financial cost, as it has already been established that expenditure on security will often be rejected in favour of improving aspects more directly related to patient care. This effectively means that hardware protection devices would not be adopted on a large scale, and dictates that a software-based security system may be the most favourable route.

It must also be recognised that even with satisfactory authentication, the issue of insider abuse is not resolved. Normal solutions to this problem are to incorporate measures such as access control and auditing. However, both of these need careful consideration if they are to be implemented effectively. The potential of auditing in particular may be wasted if handled incorrectly (as will be discussed in section 6.4).

What is therefore advocated is the use of real-time intrusion monitoring and user supervision techniques, that would combine the key elements of authentication and auditing, to provide transparent supervision of all user and system activities. This is viewed as a very good example of a security system that may be added as an overlay to operational healthcare systems, as it may be installed alongside existing security measures whilst at the same time compensating for some of their deficiencies.
However, before discussing the concept in any detail it is first necessary to show that systems abuse does actually occur in healthcare and that there is consequently a legitimate role for these techniques.

6.2 Intrusions in healthcare systems

In order to justify the need for, and advantages of having, some form of intrusion monitoring system in healthcare, this section highlights some cases of known abuse, along with general statistics relating to the frequency with which they occur.

Some brief examples of specific abuse incidents that have occurred in healthcare establishments in recent years are given below, illustrating a variety of undesirable consequences that can result.

- Three machines at European Organisation for the Research and Treatment of Cancer (EORTC) in Brussels were penetrated by a hacker and then subsequently accessed on 25 separate occasions. The system held a database of patient details including names, addresses, test results and life expectancy (The Guardian. 1993).

- A masquerade attack (involving an unnamed establishment) was reported in which a nurse hacked into a system using a memorised password belonging to a doctor. The individual involved prescribed potentially lethal drugs for one patient and altered treatment records for others. Luckily, the changes were spotted by another nurse before the drugs were administered (Audit Commission. 1994).
• Another unnamed establishment was affected when a series of computer games were installed in the system area of a PC boot disk. The presence of the games served to corrupt the disk, which ultimately led to the corruption of a months' worth of data, which then had to be reconstructed (Audit Commission. 1990).

• A local health services authority in London was affected when a routine letter inviting women to have smear tests for cervical cancer was altered by a hacker to include an obscene message. The letter was subsequently distributed to some 5,000 women in the area before the modification came to light (Computer Weekly. 1994).

• In 1989, 26,000 floppy disks purporting to contain information on AIDS were distributed to individuals and establishments on a world-wide basis. In actual fact, each disk contained a malicious Trojan Horse program and victims were subsequently informed that their hard disks would be damaged unless $378 was sent to a Post Office box in Panama (when initially installed the program modified the PC's AUTOEXEC file, and every time this was subsequently run a count was updated in a hidden file. After a random number of iterations the names of the files on the hard disk were encrypted and the files hidden. The only non-hidden file contained the request for payment). The perpetrator was eventually jailed for two-and-a-half years (Computing. 1993).
In most other organisations / sectors the principal impacts of such incidents will normally be felt in terms of financial loss and disruption. However, in healthcare various other factors will be equally, if not more, important (e.g. impacts on patient confidentiality, safety and trust).

At a more general level, the summary findings from the most recent of the two UK Audit Commission surveys cited make very interesting reading in terms of their implications for healthcare and the principal points are summarised below.

Amongst the main sectors surveyed other than healthcare were local government, education, finance, manufacturing, retail, IT and communications. The number of abuse incidents reported in the healthcare field (i.e. 127 cases) was more than for any of the other sectors, with the exception of local government (which had 193 incidents), and represented 24% of the total abuse cases reported. This can be contrasted with only 18 incidents (equating to 10% of the total number) being reported in healthcare in the previous Audit Commission survey in 1990.

A total of 334 HCEs responded to the survey, with 35% reporting some kind of abuse incidents. These are broken down as follows :

(a) 5 reported incidents of hacking (more than any other sector surveyed);
(b) 69 incidents of virus (more than any other sector except Local government);
(c) 11 incidents of fraud;
(d) 23 incidents of illicit software;
(e) 7 incidents of private work;
(f) 7 incidents of theft of data or software (again more than any other sector).

These statistics, and the magnitude of incidents in comparison to other sectors, seem to indicate that healthcare appears to be one of the more attractive areas to both internal and external abusers. They can, therefore, be added to the observations from section 3.2 to further underline the need for appropriate countermeasures.

A further, and final, illustration of the seriousness with which abuse issues are taken in healthcare is that the UK NHS has introduced a seminar specifically dedicated to the risks from viruses and hacking as a principal topic within its Information Systems Security Awareness programme (Barber et al. 1993c).

It is believed that the combination of these points provides sufficient evidence that intrusion monitoring would be appropriate in healthcare systems.

6.3 An overview of Intrusions and Intruders

The aim of this section is to provide a foundation for further discussion by describing, in general terms, what might constitute an intrusion and who might commit one.

At the highest level, intrusions or malicious activity will be the result of actions by users or processes, which will operate on one or more targets (which may include data (files), system devices and other users or processes).
The purpose of introducing supervision will be two-fold:

1. to ensure that systems are only accessed by authorised users;
2. to ensure that systems are only used for authorised purposes.

At the highest level user actions can be categorised as being either legal or illegal. However, it is useful if a more detailed breakdown than this can be derived for the different potential classes of illegal activity. For example, all of the following scenarios represent types of illegal activity that should be monitored:

- an illegal action that is still within the normal authorisation of a legitimate user (i.e. abuse of privileges);
- an action by a legitimate user which is outside the normal limits of authorisation;
- any action by an unauthorised user.

In addition, it is necessary to recognise differences in the types of potential system abuser. These have already been comprehensively categorised by Anderson (1980), and are described in table 6.1.
Chapter 6: Improving system security in healthcare

<table>
<thead>
<tr>
<th>Abuser Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>External Penetrators</td>
<td>Outsiders attempting or gaining unauthorised access to the system.</td>
</tr>
<tr>
<td>Internal Penetrators</td>
<td>Authorised users of the system who access data, resources or programs to which they are not entitled.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Masqueraders</strong> Users who operate under the identity of another user.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Clandestine users</strong> Users who evade access controls and auditing.</td>
</tr>
<tr>
<td>Misfeasors</td>
<td>Users who are authorised to use the system and resources accessed, but misuse their privileges.</td>
</tr>
</tbody>
</table>

Table 6.1: Categories of system abuser

These groupings are considered appropriate for describing the different types of user-related abuse within an intrusion monitoring framework and will, therefore, be adopted for the remainder of the discussion. Whilst it is also possible to develop a deeper profile of potential intruders, by considering factors such as the common motivations behind abuse (e.g. money, ideology, egotism etc.), these are not explored here as knowledge of them would not contribute to the process of detection.

It should be noted that Anderson’s categorisations fail to take into account any of the categories of abuse that may result from process activity (e.g. viruses, Trojan Horses etc.). This is understandable given that his analysis was made in 1980 before such incidents had become commonplace. However, there has been a significant increase in such attacks in recent years (with a number of high profile incidents, including the Internet worm, the AIDS Trojan Horse and the Michealangelo virus) and evidence suggests that viruses are now the major cause of security breaches in both networked and standalone PC systems (National
Computing Centre 1994). It is now extremely unlikely that the problem will ever totally disappear and, therefore, countering such activity should also be within the scope of a comprehensive monitor. As a consequence, a further category of intrusion, which will be called *malicious process*, will be added to Anderson’s list. These may introduce various undesirable consequences, particularly in the healthcare environment, including the alteration or destruction of data, creation of false data, degradation of system performance, crashing of systems or other effects that might render data or systems inaccessible.

The characteristics of the different intrusion groups will now be briefly examined, identifying the aspects that set them apart in terms of detection.

An *external penetrator* will best equate to the traditional perception of a “hacker”, representing someone without a legitimate purpose who should consequently play no role in the system. These should theoretically be the easiest intruders to identify, for example, by looking for signs of “exploration” or unfamiliarity with the system, as well as departures from normal user behaviour. Although reported evidence suggests that the number of “hacking” incidents are relatively small, the majority of reported cases have resulted in serious losses for the organisation involved (National Computing Centre 1994).

*Masquerader* intrusions use the compromised accounts of other users and may again be identifiable based upon a departure from the established behaviour of the masquerade victim. However, some measures (such as typical activity) may be less effective as the impostor will often be masquerading with the intention of accessing the same facilities as the legitimate user.
Chapter 6: Improving system security in healthcare

Clandestine user intrusions will primarily rely upon low-level techniques to effect a compromise (e.g. direct memory manipulation) and therefore a first step should be to restrict unnecessary access to utilities and tools that allow these techniques to be employed (e.g. assemblers, compilers etc.). Where access is required, the usage should be closely monitored (i.e. by the intrusion monitor). In general, supervision will need to operate at a lower level if such intrusions are to be identified, with particular attention to any events that may be indicative of attempts to disable or evade security controls.

Misfeasors will again be more difficult to identify, as they have legitimate access to the system. However, as with a masquerade, unauthorised activities should stand out in some way when compared to typical patterns of use (obviously assuming that the majority of use is for authorised purposes). Misfeasors often behave outside the rules of the system in general and examples of their activity may include:

- illegal storage of information;
- illegal use of applications;
- illegal dissemination of information to unauthorised parties;
- games playing.

Finally, with regard to Malicious processes, various different categories can be identified (Brunnstein et al. 1990), as briefly described in the pages that follow.
• **Viruses**

These are self-replicating programs that carry a (potentially destructive) "payload". Viruses may "infect" disks and / or individual files, spreading via network communications and exchange of diskettes (Hruska 1992).

• **Worms**

Also known as "rabbit" programs, these are self-replicating programs designed to "breed" within the system, spreading in the same manner as a virus but lacking the "payload" element. Whilst not necessarily designed to be malicious these may still pose a threat to security by consuming system resources and degrading performance (Denning 1990).

• **Trojan Horses**

Taking their name from the famous Greek horse, these are executable programs that claim and / or appear to perform some useful or harmless function, but also conceal a malicious purpose (e.g. stealing passwords, corruption of files). Trojans may be introduced onto a system in the form of new programs or in modified versions of existing applications / utilities (Parker 1990).

• **Logic and time bombs**

*Logic bombs* and *time bombs* both refer to malicious code embedded within a program that is only activated (or triggered) when some specific condition is fulfilled. Logic bombs are triggered by the occurrence of a specific event (or
event series) within the system. A classic example is the modification of a payroll system by an employee to monitor for the removal of his / her name, with encryption of all company data programmed to occur if this ever happens. By contrast, time bombs are programmed to trigger either after a certain period of time has elapsed or when a specific time / date is reached (Hruska 1992).

All of the above may be disguised in various ways and the nature of the malicious activity will vary (e.g. some may simply display a message, whilst others crash the system, delete files, encrypt data etc.). For example, the complete identification / analysis of a virus involves knowing the following (Gold 1989):

- the mechanism by which it creates its effects;
- the mechanism by which it conceals its existence;
- the mechanism by which it replicates;
- any clues it gives to its existence;
- what its effects actually are (i.e. the payload).

Therefore, without knowing exactly what signs to look for it is only possible to monitor for the broadest indicators - which may not be effective in all cases. In general it may be easier to detect the introduction of the malicious process (i.e. the "infection") rather than the resulting payload action. A further difficulty arises from the fact that each new generation of malicious process may be explicitly designed so as to avoid existing means of detection (e.g. the emergence of self-mutating viruses).
Referring back to the list of Audit Commission figures from section 6.2, it can be seen that cases (a) and (b) would be respectively classified as penetrator and malicious process related intrusions, whereas in cases (c) to (f) it is likely that a large number would almost certainly be classifiable as misfeasar activity.

### 6.4 Traditional approaches to Intrusion Detection

In most IT environments, including healthcare, details of intrusions are traditionally captured by the system audit trail. However, in many organisations auditing does not fulfil its full potential and is viewed with low regard by systems administrators. The key reasons for this include:

- the large volume of data collected (even in small or medium sized systems) and the associated burdens terms of analysis and storage space consumed. This can be illustrated by considering the findings of Piccioto (1987) based on the implementation of auditing at Mitre Corporation in the USA. On average, auditing of workstation activity was found to generate around 7MB of data per day (24 hours), rising to 136MB under peak / worst case conditions;

- the further overhead, in terms of staff time and effort, required to inspect data and follow-up anomalies;

- the actual interpretation of the data and identification of an intrusion may be difficult (especially if the intruder has attempted to cover-up any activity);
Chapter 6: Improving system security in healthcare

• a lot of the data collected will not be security relevant, but must still be analysed to identify that which is;

• the fact that audit trail analysis only occurs after events have occurred (by which time damage may already have been done).

An example of the typical attitudes towards auditing is presented in a discussion of the “Green System” security architecture used in the Danish health sector (Birkegaard 1990). Here the author states that the following up of audit trails generated by the system is a matter of the security administrators “personal taste” and that, whilst records will not always be examined, “if he needs them, he can use them”. This, of course, fails to acknowledge that an administrator might only think he needs the records if a security breach is already suspected.

The lax attitude towards auditing is further illustrated in figure 6.1, based upon the results obtained in another general survey of computer abuse (Gliss 1990). This information relates to auditing in mini / mainframe systems (using audit trails, log files and monitoring print-outs) and reveals that only 10% of organisations always follow-up their reports. The situation in the PC environment was even worse, with only 12% of systems even having any auditing / control software installed. Given these statistics, it is unsurprising that of the total abuse incidents covered in the latest Audit Commission survey, over half were only actually detected by accidental means.
Considering the feedback received from healthcare personnel and the SEISMED survey it would appear unlikely that the healthcare environment is any different in this respect or that the situation has changed significantly.

It is envisaged that an intrusion monitoring system operating in real-time would overcome the problems identified, analysing the significance of data as it is obtained and thereby having the potential to identify intrusions before any major security compromise occurs.

The sections that follow present a set of outline ideas and preliminary designs for a real-time Intrusion Monitoring System (which will herein be referred to as IMS) to meet these objectives.
6.5 Advanced approaches to Intrusion Detection

It is envisaged that intrusion detection could be improved considerably beyond the level of simple audit trailing by building it into a more comprehensive framework that encompasses full user authentication and supervision. This section will introduce the various approaches that would be desirable within IMS. It should be noted that many of these techniques could be equally applicable outside the healthcare sector and, for this reason, the majority of the discussion is pitched at a general level.

It is suggested that IMS intrusion detection could be based on a combination of several independent strategies:

- auditing of the local system configuration;
- initial user identification and authentication;
- on-going comparison of user activities against historical “behaviour profiles”;
- use of generic rules to identify potentially anomalous system events.

The integration of these elements into a basic user supervision strategy is shown in figure 6.2 overleaf. It is believed that this structure has the potential to provide a very comprehensive protection framework and each of the component factors will now be described in more detail.
6.5.1 System auditing

The first task of IMS should be to ensure the integrity of the system upon which supervision is to be conducted. The local system should therefore be checked at user login or system start-up time to ensure that no changes have been made that may compromise security.

A number of stages may be incorporated into start-up tests, including:

- auditing of the basic system set-up and hardware configuration (e.g. processor type, operating system version, default access paths etc.);
- checking the integrity of important system files (e.g. login scripts, application set-up files);
- virus scanning.
Beginning with the first of these points, it is envisaged that certain configuration changes may have serious implications from a security standpoint (where the configuration in this context encompasses factors relating to the system hardware, the operating system and any significant user defined settings). For example they might:

1. be indicative of physical tampering with (or theft of) equipment.
2. affect the compatibility and/or performance of existing applications (including the IMS supervisor itself), which could result in accidental security compromise;
3. be indicative of a deliberate attempt to compromise security.

As a countermeasure, relevant configuration data should be collected and stored by IMS, which may then be used for comparison against the system configuration on subsequent occasions to ensure that everything is still as expected. For example, in the case of a PC system, various elements of the configuration may be audited and a selection are presented in table 6.2, indicating the classes of compromise that the monitoring of each would help to highlight.

Although data could also be collected relating to a number of other aspects of the PC environment (e.g. video mode, number of printers, number of serial ports etc.), it would serve little purpose as any changes would not have significant implications for the secure operation of the system.
Chapter 6: Improving system security in healthcare

<table>
<thead>
<tr>
<th>Configuration Characteristic</th>
<th>Physical alteration / theft</th>
<th>Compatibility / performance problems</th>
<th>Security compromise attempt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total conventional RAM size</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Free conventional RAM</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>BIOS release date</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOS version number</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of DOS files</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Default working directory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of floppy drives</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of hard drives</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard drives capacities</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maths coprocessor type</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control-break setting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command processor specification</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Path specifications</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.2: Auditable characteristics of PC configuration

Whilst it is acknowledged that various public domain utilities and some commercial products already exist to collect configuration data (e.g. "CheckIt Analyst" (S&S International 1994)), the purpose of most of them is simply to report the configuration or highlight changes that may affect application compatibility. Beyond this, the fuller implications from a security standpoint are not considered.

With regard to file integrity, the modification of certain files could adversely affect the performance of the system or compromise security (e.g. again using the PC environment as an example, maintaining the integrity of the AUTOEXEC.BAT and CONFIG.SYS files may be particularly important). Integrity can be ensured by calculating checksums based on the file content (Simmons 1992). The IMS system would validate each file during start-up by
recalculating the checksum and comparing it with the value already held. A different result would be indicative of file modification and could be used as a trigger for further action.

It is envisaged that the final aspect of the audit, that of virus scanning, could be achieved by incorporating a link from the IMS into one of the many existing anti-virus utilities that are available.

Whilst this element of IMS would not provide intrusion detection in real-time, it is nevertheless important to provide such a measurement of system integrity every time it is first accessed.

6.5.2 Initial user identification and authentication

Identification of the current user is necessary at the start of a session to enable the system to determine which profile should be used for supervision. In theory, the subsequent monitoring of behaviour could then act as the mechanism for authenticating the claimed identity. However, the inclusion of an initial authentication phase would provide some basis for believing that the correct identity was given from the outset, which would allow the system to proceed with an initial high confidence of user legitimacy (lessening the chances of an external penetrator or masquerader class intrusion). Some consideration of appropriate authentication mechanisms is, therefore, necessary.

At the most basic level, the simple passwords could be used for this purpose (as in most existing healthcare systems). In this sense, the procedure would then remain much the same
as a conventional system login. However, two factors suggest that traditional password-based methods may no longer afford sufficient protection:

- the level of data sensitivity in many healthcare systems adds weight to the argument that passwords (which often provide a weak/unreliable basis for authentication anyway (Jobusch and Oldehoeft 1989)) should be supplemented by other mechanisms;

- in real terms, passwords do not provide a particularly user-friendly authentication mechanism, in that the burden of proof is placed upon the user. Users often have difficulty remembering passwords, leading them to write them down or use ones that could be easily guessed - negating the security benefits. Things get especially complex where users must remember several passwords (for example, where secondary levels of passwords are used to access specific applications or data, or where users have accounts on several systems). Even in cases where users have no trouble remembering passwords, it is likely that many would rather not have to do so.

So, whilst various means exist to enhance the security of traditional passwords (see, for example, Gritzalis et al. 1992), the second of these points still highlights a potential disadvantage.

With regard to alternatives, the use of smart card systems may have a place in overcoming these problems (Zoreda and Oton 1994), but may not be practical as a compulsory measure
as this would introduce an immediate financial burden across the whole system (e.g. for the
installation of card readers and issuing of cards), which most HCEs would not be able to
tolerate at the present time. However, the option of advanced authentication is not entirely
precluded.

In some cases, it may be possible to take advantage of one of the previously identified
trends in modern healthcare - namely the use multimedia systems. These may allow several
new options to be introduced for improving authentication. For example, appropriate
hardware for implementing several biometric identification methods may already be present
"as standard" in a multimedia configuration (e.g. cameras which may be present for video
conferencing purposes could also be used for image / “faceprint” recognition; microphones
and audio processing facilities could be used as the basis for voice recognition). These
techniques have been successfully implemented elsewhere, delivering adequate
authentication performance and gaining a high degree of user acceptance (Sherman 1992).
As such they should integrate well with multimedia healthcare systems, providing a more
user-friendly method than most in current use. However, the presence of such hardware
enhancements should not be a prerequisite of the authentication strategy for the same
reasons as smart cards. Nevertheless, some mechanism of intelligent supervision would be
desirable that can allow such extra facilities to be utilised if they are present.

It is acknowledged that there are a number of other biometric authentication measures that
may also be technically feasible, including fingerprint, hand geometry or signature
recognition. However, none of these really offer any better basis for authentication and, in
actual fact, hold less potential for transparent or continuous integration into the supervision
system given that they require more specific actions on the part of the user (U.S. Congress 1987). In addition, the required hardware in each of these cases would not be a likely "standard" feature of any healthcare system (multimedia or otherwise) and would, therefore, represent an additional expense.

All user authentication information (be it passwords or the more advanced vocal / facial biometric information) could be incorporated as the first aspect of a user personality profile. The other aspect, as indicated in figure 6.3 below, relates to details of user behavioural activity that will be discussed further in the next section.

6.5.3 User behaviour profiling

It should be possible to detect intrusions related to user activity by comparing the current system use against established patterns of user behaviour (i.e. profiles) and then looking for anomalies. In this way, any activity that is not compatible with the normal behaviour of the user in question can be highlighted as a potential cause for concern. Such an approach was originally proposed by Denning (1987) and was the basis for the IDES system, which will be discussed later in this chapter.
Over time users become familiar with the way that their system normally operates and will notice any significant departures from the norm. Similarly, experienced system administrators are often able to detect anomalies from the way the system appears to be running or by monitoring user activity. This is illustrated by the following quote, taken from the book “Cyberpunk” (Hafner and Markoff 1991), describing system administrator Steph Marr’s initial discovery of a hacker:

“Marr was one of the people who worked to keep Santa Cruz Operation’s network of computers up and running. He had been there for a year, long enough to know that certain users not only had certain privileges on the system but also had individual habits. Engineers logged on from their homes late at night; secretaries logged on only from work and only during working hours... one of the secretaries who used the computer was acting out of character. She was logging in after hours, cruising the system and trying to peek into other people directories...”

The purpose of behaviour monitoring is to allow this anomaly detection ability to be encapsulated within the IMS. The maintenance of behaviour profiles for each legitimate user would enable their activities to be compared against what is historically normal.

It is envisaged that behaviour profiling could be usefully implemented at two levels: user class (high level) and user specific (low level). These approaches are described in the sections that follow.

6.5.3.1 User class profiling

At this level profiles are maintained describing the expected behaviour for each class of user. This is based on the premise that it is possible to separate users into different
behavioural classes according to their role within the organisation and then develop general-level profiles of acceptable activity within each group.

Validation of current activity against this profile will give an indication of whether users are operating within their legitimate bounds (i.e. providing a good means of detecting misfeasors, as well as any penetrators / masqueraders who stray from the accepted behaviour of the user being impersonated). However, this approach is not sufficient in terms of authentication as users of the same class would be able to successfully masquerade.

The approach is heavily dependent upon the personnel, applications and responsibilities within each specific environment. Each organisation would need to determine its own user classes and then define appropriate rules for behaviour within them (a task which would be performed by the System Security Manager, or similar). Even where the same user classes existed in different establishments it is unlikely that the behaviour profiles would be the same. For example, a “secretary” in one establishment may be permitted to use significantly different systems and applications from someone else performing the same role elsewhere. As such the behaviour profile for the user class “secretary” would be equally different. Once developed, however, it is anticipated that the behaviour rules for each class would remain relatively static.

It may be noted that high level profiling of this type represents a similar activity to the development of an Access Control Matrix. Indeed, some operating systems would allow the “behaviour rules” derived for each class to be used to explicitly define access controls
rather than requiring a supervision system to detect deviations. As such the usefulness of maintaining class behaviour profiles is reduced in these contexts. An example of a user class profiling exercise that has been conducted within a local HCE is given in section 6.6.

6.5.3.2 User-specific profiling

At a lower level, profiling may be based upon historical patterns of behaviour for each individual user. This requires the maintenance of more detailed information but, as it is user specific, will also allow a capability for continuous user authentication to be incorporated into the supervisor.

Many activities can be identified that may provide a suitable basis for user-specific profiling and previous research (Lunt at al. 1989) has differentiated between methods on the basis of whether they provide discrete or continuous indicators, as defined below:

- **Discrete indicators**

  Provide single measurements that are obtained once during a session (normally at the beginning) and may be analysed immediately by the intrusion monitor;

- **Continuous indicators**

  Provide information throughout a session, which changes according to the types of activity in progress (and must, therefore, be periodically reassessed).

For consistency, the same terminology will be used here.
Individual behaviour profiles would need to be developed using data collected over a reasonably long time period, in order to establish what constitutes "normal" behaviour for each legitimate user. However, unlike class profiles, this information could be collected and analysed by IMS without the need for human involvement. In addition, the system could take account of legitimate changes in the subject's behaviour, with some form of profile refinement being incorporated.

Table 6.3 lists a series of characteristics that could potentially be used as a basis for user identity and activity assessment. A brief outline description is given in each case, along with the perceived advantages and disadvantages of each approach. The second part of the "description" column is intended to indicate whether the detection method would provide a discrete (D) or continuous (C) measurement. For completeness, table entries are also included for System Auditing and Generic Rules detection methods, although neither are directly related to user behaviour.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keystroke Analysis</td>
<td>The authentication of users based upon distinctive characteristics of their typing style / rhythms. May be based upon statistics such as inter-keystroke time, keypress duration, frequency of typing errors etc. In this case, analysis uses any arbitrary text input. The approach is described in detail in chapter 8.</td>
<td>C • Authentication. • Based on behavioural characteristic (difficult to imitate or transfer to others). • Can use any text as basis for analysis.</td>
<td>• Only suited to external penetrators or masqueraders. • Still allows a &quot;window&quot; for abuse (i.e. some keystrokes may be entered before detection). • Only useful in text intensive contexts. • Requires large test sample. • Requires intelligent terminal / device to collect timings.</td>
</tr>
</tbody>
</table>

Table 6.3: Potential IMS profile characteristics
### Table 6.3: Potential IMS profile characteristics (continued)

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
</table>
| **Keystroke Analysis** | **(Static)** As above, but based on analysis of specific text strings that subject will enter regularly (e.g. passwords, OS commands). Normally used to provide a discrete judgement. | **D**  
  - Authentication.  
  - Based on behavioural characteristic (difficult to imitate or transfer to others).  
  - Less intensive sampling.  
  - Samples concentrate on specific text (which is more familiar to subject). | **C**  
  - Heavily context dependant (can only test for specific text).  
  - Requires intelligent terminal / device to collect timings. |
| **Access Time**      | Time(s) between which subjects typically access IT systems. In some cases there may be a detectable correlation between access time and application usage, allowing a continuous measure. | **D**  
  - Effective for users with strict working hours.  
  - Various activities may only be seen as “normal behaviour” if they occur within certain periods, and would classed as anomalous at all other times. | **C**  
  - Valid access times may vary significantly with some classes of user.  
  - The percentage of cases where a correlation can also be made to application usage may be very small, as few users are likely to exhibit such regimented behaviour. |
| **Access Location**  | May be approached from two perspectives: monitoring the location(s) from which subjects typically access IT systems OR monitoring which subjects normally access from any given terminal / port. | **D**  
  - Many users only access from specific terminals / locations, highlighting use of other locations as anomalous. | **C**  
  - Some users may legitimately require access from many locations. |
| **OS Command Usage** | Type and frequency of operating system commands used.                       | **C**  
  - Will differentiate between subjects of different expertise. | **D**  
  - Not applicable in Graphical User Interface (GUI) environments. |
| **Application Use**  | Type and frequency of application systems used.                             | **C**  
  - Most users will spend most time using only a small number of applications.  
  - Significant use of an infrequently used application may be suspicious. | **D**  
  - Masqueraders would probably target the same applications as the user they are impersonating. |
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Interaction</td>
<td>Monitoring of the method(s) by which a subject commonly interacts with the system / applications (e.g. keyboard or mouse, commands or menus).</td>
<td>C • Likely to give an indication of users experience and / or familiarity with a system (e.g. whilst novice users could be expected to rely on a mouse &amp; pull-down menus, users with more experience might make use of short-cut key sequences etc.).</td>
<td>• Not a particularly strong discriminating measure in its own right.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Many users may be inconsistent in the methods they use.</td>
</tr>
<tr>
<td>File Access</td>
<td>Data files most frequently / recently used by each subject and for what purpose (i.e. read, write).</td>
<td>C • Anticipated that most access occurs within a small “working set” of commonly used files.</td>
<td>• Natural that some files will be accessed infrequently or irregularly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Masqueraders may target the same files as the legitimate user.</td>
</tr>
<tr>
<td>Resource Usage</td>
<td>Statistics of typical usage of system resources (e.g. CPU, memory, disk) associated with each subject.</td>
<td>C • Significant changes in user activity should result in different statistics.</td>
<td>• Provides a rather loose measurement.</td>
</tr>
<tr>
<td>Login Failure</td>
<td>Tracking of unsuccessful attempts to gain system access.</td>
<td>D • Valid users should only require a small number of attempts (e.g. a maximum of three).</td>
<td>• Detects no anomaly if intruder knows the password etc.</td>
</tr>
<tr>
<td>Access Violations</td>
<td>Tracking of the number of access violations (e.g. to files, data, applications, devices) made by a user / process during a session.</td>
<td>C • Provides a good indicator of unauthorised behaviour.</td>
<td>• Clandestine users may bypass access controls.</td>
</tr>
<tr>
<td>Session Length</td>
<td>The duration of a user session. May be monitored in terms of either a discrete measure of the overall session length or as a continuous measure past a certain threshold.</td>
<td>D C • May be useful in conjunction with Access Time to detect unusual periods of activity.</td>
<td>• A discrete measure would only be available after session termination (of little use to prevent damage).</td>
</tr>
<tr>
<td>Network Traffic Analysis</td>
<td>Characteristics of the users network usage, based on the monitoring of data on the network lines (e.g. packet size, frequency, source and destination).</td>
<td>C • May contribute to the identification of all classes of user intrusion.</td>
<td>• May require a hardware device to monitor network activity and collect data.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Complex analysis.</td>
</tr>
</tbody>
</table>

Table 6.3: Potential IMS profile characteristics (continued)
**Table 6.3 : Potential IMS profile characteristics (continued)**

In addition to these, there may also be potential in assessing a number of more minor behavioural measures or statistics associated with user sessions. These might include:

- number of files created or deleted;
- number of print jobs submitted or pages printed.

However, the usefulness of these measures is envisaged as being much more context-dependant than those listed in the table and, therefore, their monitoring may not be worthwhile in terms of the additional processing and storage requirements that would be incurred.

It should be evident that some of the methods listed in the table will provide much stronger indicators than others. It is in fact possible to break the potential elements of the personality / behaviour profile down into different levels of effectiveness as shown in figure
6.4. For simplicity, only a subset of the measures identified are shown and they are split into just three levels (although there could conceivably be more in practice).

As indicated in the figure, the strength of the measures in terms of their potential for accurate user authentication decreases as one moves down through the levels. However, at the same time, other positive factors can be cited, including:

- ease of practical implementation / integration into existing systems;
- transparency of the measure;
- potential to detect abusers other than penetrators;
- financial viability.

However, it is expected that in most cases no single factor would provide an adequate detection measure and the combination and analysis of several would be most appropriate.
For this reason the system will require some ability to identify *aggregations* of activity that could lead to a security violation.

In fact, the methods vary in terms of their ability to detect the different types of intrusion. Many of them, for example "keystroke analysis", represent "authentication only" measures and would only be suitable to distinguish between a legitimate subject and an impostor, whereas "file access" could also be used to identify misuse by a valid system user and, therefore, allow more comprehensive supervision. Table 6.4 shows which methods are appropriate to detect each type of abuser.

<table>
<thead>
<tr>
<th>Method</th>
<th>External Penetrator</th>
<th>Masquerader</th>
<th>Clandestine User</th>
<th>Misfeasor</th>
<th>Malicious Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keystroke Analysis</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Dynamic)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keystroke Analysis</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Static)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Access Time</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Access Location</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS Commands</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Application Use</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>User Interaction</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>File Access</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Resource Usage</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Login Failure</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Access Violations</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Session Length</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Traffic Analysis</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>System Auditing</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Generic Rules</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 6.4: Scope of intrusion detection methods
Chapter 6: Improving system security in healthcare

The applicability of each measure will also vary between different users (e.g. monitoring of operating system commands will not be applicable to a user who is not granted command line access). The IMS would, therefore, need to allow the administrator to determine which measures will be active on which accounts.

6.5.4 Generic intrusion indicators and rules

In some cases intrusions may be identified without requiring any historical knowledge of specific users behaviour. Rules may be incorporated into the system to allow it to spot specific events (or event series) that may be indicative of a security compromise. Relevant activity data may be compared against these to see if it might form part of an intrusion scenario. This will assist in the monitoring of the system state as well as user-related activity.

The “intrusion rules” could be based upon a number of factors, as listed below:

- known intrusion scenarios / patterns of abuse (also known as “attack signatures”)
  - see table 6.5;
- known weaknesses of the host system (e.g. operating system vulnerabilities);
- HCE security policy (e.g. rules for data / file access by different staff groups);
- advice from security experts;
- audit trail analysis (Leipins and Vaccaro 1989).
The examination of literature relating to known intrusion scenarios and documented hacking case studies (Landreth 1985; Stoll 1988; Hafner and Markoff 1991; Sterling 1992; Quittner and Statalla 1995) reveals several classes of event that should at least be regarded as "suspicious". A series of examples, drawn from both these sources and other knowledge, are given in table 6.5 below, including the type of event, a brief description and the type(s) of intrusion that may be indicated (along with a general confidence rating - low, medium or high). Note that some of the characteristics that are monitored in behaviour profiles (e.g. access time) may also be incorporated at this level as more general rules.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Potential Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consecutive login failures</td>
<td>Consecutive or frequent failed attempts to access the system (or a specific account) may indicate someone trying to guess their way in.</td>
<td>External penetrator Masquerader H</td>
</tr>
<tr>
<td>Consecutive access violations</td>
<td>A significant number of failures during a session indicates that the user may be trying to access objects / resources for which he / she is unauthorised.</td>
<td>Penetrator, Miseason H</td>
</tr>
<tr>
<td>Guest / anonymous access</td>
<td>Guest accounts and other forms of anonymous access can provide a &quot;foothold&quot; for hackers and any significant use of such accounts should be monitored.</td>
<td>External penetrator L</td>
</tr>
<tr>
<td>Account overuse</td>
<td>Unusually high levels of activity on user accounts may be suspicious. Simultaneous sessions utilising the same account may indicate that a penetrator is using the system.</td>
<td>Penetrator M</td>
</tr>
<tr>
<td>Excessive session length</td>
<td>An unusually long session may indicate that the system is being misused.</td>
<td>Penetrator, Miseason L</td>
</tr>
<tr>
<td>Out of hours access</td>
<td>Out of hours access (especially at night) may indicate unauthorised activity.</td>
<td>Penetrator, Miseason L</td>
</tr>
<tr>
<td>Access of infrequently used file</td>
<td>Access to a file that is seldom used may be indicative of an unauthorised user browsing through the system.</td>
<td>Penetrator L</td>
</tr>
<tr>
<td>Modification of login or system configuration files</td>
<td>Login and configuration files can control important aspects of system behaviour and will not normally be changed on a frequent basis. Modification may assist a penetrator in compromising the account or system.</td>
<td>Penetrator M</td>
</tr>
<tr>
<td>Copying of password file</td>
<td>Whilst many systems permit read access to the password file, attempts to do so should be regarded as suspicious. Penetrators in possession of the file would be able to run password cracking software against the contents.</td>
<td>External penetrator H</td>
</tr>
</tbody>
</table>

Table 6.5 : Examples of generic intrusion indicators
## Chapter 6: Improving system security in healthcare

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Potential Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copying of system files</td>
<td>Duplication of operating system files should not normally be necessary. These are standard targets for malicious users wishing study / modify the system and exploit weaknesses.</td>
<td>Penetrator</td>
</tr>
<tr>
<td>Use of inactive accounts</td>
<td>Sudden or unexpected activity on accounts that have been dormant for long periods.</td>
<td>External penetrator Masquerader</td>
</tr>
<tr>
<td>Repeated / frequent use of “WHO”</td>
<td>Repeated checking to see who else is logged into a system may be an indication of a penetrator “watching his / her back”. Hackers will normally look out for the system administrator being logged in.</td>
<td>External penetrator</td>
</tr>
<tr>
<td>Extensive use of “help” systems</td>
<td>External penetrators may be unfamiliar with the system and its facilities and may refer to help systems frequently.</td>
<td>External penetrator</td>
</tr>
<tr>
<td>Repeated / frequent user errors</td>
<td>External penetrators may highlight their unfamiliarity with systems through errors such as issuing commands that do not exist, and / or attempting to access non-existent files or directories. Errors above a certain threshold will be suspicious.</td>
<td>External penetrator</td>
</tr>
<tr>
<td>Accessing of dummy (“bait”) files</td>
<td>This involves setting deliberate “traps” for unauthorised system browsers in the form of specially monitored dummy files which should appear to be interesting. Users with legitimate access to the system / area should be instructed to ignore these and, therefore, any access attempts indicate a potential intrusion.</td>
<td>Penetrator</td>
</tr>
<tr>
<td>Excessive memory / file space consumption</td>
<td>Consumption of unusually large amounts of memory or storage by a user or process may indicate undesirable activity (e.g. a hacker accumulating files, a worm duplicating in the system).</td>
<td>Penetrator, Malicious process (Worm)</td>
</tr>
<tr>
<td>Remote use of system accounts / privileges</td>
<td>Anticipated that authorised system management / “superuser” account holders would most commonly operate from local terminals. Any access of such accounts from remote locations should be noted.</td>
<td>External Penetrator</td>
</tr>
<tr>
<td>Use of local system as a gateway to other hosts</td>
<td>A remote login to the system, followed by an attempt to connect to another remote system may indicate a penetrator or process passing through (“network weaving”).</td>
<td>External penetrator, Malicious process (worm, virus)</td>
</tr>
<tr>
<td>Modification of “system” memory areas</td>
<td>Alteration of certain memory locations may provide a means to bypass security controls and may interfere with system operation.</td>
<td>Clandestine user, Malicious process</td>
</tr>
<tr>
<td>Disk boot sector modification</td>
<td>Many viruses infect systems by installing themselves in the disk boot sector that is read during start-up. As such, attempts to modify boot sectors should be monitored.</td>
<td>Malicious process (virus)</td>
</tr>
<tr>
<td>Increase in number of “hidden” files</td>
<td>Many OS allow for “hidden” files that do not show up in normal directory listings. Addition of a hidden files may represent a simplistic attempt to disguise the introduction of malicious programs.</td>
<td>Malicious process (virus, Trojan horse)</td>
</tr>
</tbody>
</table>

Table 6.5: Examples of generic intrusion indicators (continued)
Chapter 6: Improving system security in healthcare

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Potential Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modification of executable file</td>
<td>Executable programs should remain static under most circumstances and alteration may indicate the introduction of malicious code.</td>
<td>Penetrator, Malicious process (virus, Trojan horse)</td>
</tr>
<tr>
<td>Addition / replacement of executable file</td>
<td>Addition or replacement of executable files outside a users own area (and especially in “system” areas) is unusual and may represent an attempt to introduce malicious code.</td>
<td>Penetrator, Malicious process (virus, Trojan horse)</td>
</tr>
</tbody>
</table>

Table 6.5: Examples of generic intrusion indicators (continued)

The events in the table are all related to established patterns of abusive behaviour that may relate to almost any system. As previously indicated, further rules could be devised at a later stage when the implementation platform has been selected, based upon any known weaknesses in the environment. In addition, examples of healthcare-specific anomaly indicators that might be derived from HCE policy include:

- repeated access (browsing) of different patient records;
- extensive printouts of patient data.

It may be noted that many of the indicators represent security risks that can be overcome by explicit action from system administration (e.g. disablement of “guest” accounts, limitation of access times for each account). However, many organisations may not observe these precautions and the use of such rules therefore provides a means to compensate in cases where administration is not comprehensive.

Whilst no single event may be conclusive of an intrusion, occurrences may be used to increase an IMS alert status (with events that are considered most significant causing...
greater increases). In this way, certain combinations of events may be identified that are much more significant than any event on its own (for example, the occurrence of three low-confidence events such as excessive night-time use of a "guest" account would be a very strong indication of an intrusion).

It should be noted that the larger the rule-base, the longer it will take for the system to search on each monitoring iteration (and, hence, the greater the processing overhead on the system). As such, in cases where efficiency is of paramount concern, it may be desirable to prioritise the rules along the lines of the confidence ratings shown in table 6.5. In this way, the monitoring system could minimise its effort by initially testing only the "high confidence" rules, and then only proceeding to the next level of rules if one or more of these were satisfied.

A further limitation on this aspect is that many intrusions may exhibit characteristics that, whilst easy for a user or administrator to spot as unusual, would be difficult for the supervision system to identify. For example, both of the following are quite often caused by the presence of a virus:

- unexpected slowness of system response / application operation;
- system crash.

However, in either case it would be difficult for an intrusion monitor to separate these events from other system activity and determine whether the event was a symptom of a deliberately malicious process or some other anomaly in the system.
6.6 Generic behaviour profiles for healthcare users

This section builds upon the idea of user class profiling that was previously outlined in section 6.5.3.1 and describes how the concept could be used to formulate high-level profiles suitable for the staff within a healthcare establishment.

6.6.1 Introduction

It can be observed that many of the behaviour characteristics described in section 6.5.3.2 (e.g. keystroke analysis, time and location of access, use of operating system commands) represent techniques that are equally applicable in almost any environment (i.e. not just healthcare). However, it is envisaged that supervision may be more effective using a two-tier profiling approach, with general profiles at the first level and more detailed profiles underneath. As such, an appropriate approach is the development of user class profiles, which describe the general behaviour characteristics of different types of HCE personnel.

In order to be monitored by the supervision system, only behaviour that relates to the use of information systems can be profiled (i.e. a limited "window" on the overall behaviour of different staff). The profiles, therefore, aim to categorise how each type of HCE user typically uses information systems, as well as modelling any additional information that may usefully set one class of user apart from another. In short, this will include:

- *what* systems are used;
- *when* they are used;
- *how* they are used.
Such information can be used to formulate general rules for reasonable behaviour within each of the different job categories; rules which could subsequently be incorporated into the full IMS. This approach is very similar to the idea of “role profiling” that was proposed by Calitz et al (1995) in work published after the completion of this aspect of the investigation.

It was considered that suitable profiles could be developed using the responses to a relatively small series of basic questions, obtained from a broad cross-section of HCE staff. These questions would all be relatively easy for staff to answer and, as such, a questionnaire approach was considered a sufficient data collection method (and probably more appropriate than conducting detailed interviews of individual staff).

For convenience, the study was based in the Plymouth reference centre (i.e. Derriford Hospital), with support and assistance from the Trust Information Doctor (Dr P.N.Gaunt). It should be remembered at this point that the resulting profiles were, therefore, only intended to be representative within the Derriford domain and could not be guaranteed to apply elsewhere.

6.6.2 Categorisation of HCE staff

It was originally envisaged that class profiling work would begin with more specific subdivisions of the existing General, Management and IT & Security Personnel staff groupings taken from the SEISMED work. This led to the definition of a number of generic roles as shown in figure 6.5.
However, it was quickly established that attempting to address all of these groups would be impractical within the context of Derriford Hospital due to other research projects also in progress. At the time at which this study was proposed, the staff who would have constituted the IT & Security audience had only just finished participating in a different questionnaire study, whilst the attention of HCE Management staff was required for interview studies. In view of this, it was considered unfair to burden these groups further and the hospital requested that the study be confined to the general / end-user population. This limitation was considered acceptable given that the general user category encompassed the largest number of generic roles and was also the most easily accessible of the groups within the hospital. As a result, it was considered that specifically targeting this audience offered the most scope for obtaining survey responses anyway.

Fig. 6.5: Generic Categorisation of HCE Staff
The specific focus on general staff was used as opportunity to expand the choice of generic roles with two further categories, Clerk and PAM (for Professions Allied to Medicine) having been suggested by the hospital, so that a more accurate classification could be achieved. Thus, the final staff categories used were as illustrated in figure 6.6, with the range of roles encompassed ensuring that staff would be able to select a category into which their actual role would fit reasonably naturally.

**Fig. 6.6 : Final categorisation of general HCE staff**

### 6.6.3 Questionnaire content

As previously stated, it was considered feasible to formulate the class profiles from responses to basic questions relating to the *what, when* and *how* of information system usage within the HCE.
Chapter 6: Improving system security in healthcare

The questionnaires that were distributed included eight questions designed to elicit the required information relating to HCE staff and their use of information systems. The following section presents a breakdown of the questionnaire material and explains the rationale behind each question (or group of related questions).

The questions are listed below, presented in the same manner as they appeared in the questionnaire (however, it should be noted that the questions are sequentially numbered here for reference purposes only and did not appear in this strict sequence in the final questionnaire).

1. Which of the following categories best describes your role?
   - Consultant
   - Researcher
   - PAM
   - Junior Doctor
   - Administrator
   - Clerk
   - Nurse
   - Secretary

2. What are your typical hours of work (if fixed)? From ___ to ___

3. On average, how long do you spend using the hospital computer systems each day? ____ hours?

4. Please indicate the types of computer system that you regularly use?
   - Standalone PC
   - PC on a Network
   - Terminal to Hospital Computer
   - Remote (non-Derriford) System
   - Other (please specify)

5. Please indicate which of the following types of applications you use and how frequently (1 whole of day, 2 part of day, 3 less frequently, leave blank if never used)
   - PAS
   - Radiology
   - Clinical Workstation
   - Financial Systems
   - Clinical Laboratory
   - Clinical Laboratory
   - Other (please specify)
6. Which of the following types of data do you create (C), access (A), update (U)?
Please tick all boxes that apply:

- C  A  U  Patient Care/Diagnosis
- C  A  U  Patient Administration
- C  A  U  Personnel
- C  A  U  Resource Management
- C  A  U  General Hospital Administration
- C  A  U  Financial
- C  A  U  Laboratory, Radiology or other service dept.

7. Do you normally access information systems from more than one workstation / terminal?  □ Yes  □ No
   If yes, are these workstation / terminals in different areas of the Hospital?  □ Yes  □ No

8. Do you legitimately share a group password?  □ Yes  □ No

The rationale behind each of these questions is given in table 6.6. This indicates the information that should be obtained from each question (or related group) and how it contributes to the development of class profiles.

<table>
<thead>
<tr>
<th>Question(s)</th>
<th>Reason for inclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Information intended to allow the responses to be grouped into appropriate classes.</td>
</tr>
<tr>
<td>2,3</td>
<td>Information that may be used to determine the typical / possible system access times for different classes of user.</td>
</tr>
<tr>
<td>4,5,6</td>
<td>Responses to these questions gave information on what systems are used and how. The data types listed are based upon a principal subset of the groups from the healthcare generic data model, whilst the named applications relate to the main systems available within Derriford Hospital.</td>
</tr>
<tr>
<td>7</td>
<td>Gave information on whether the location of access is relevant in monitoring a particular user class.</td>
</tr>
<tr>
<td>8</td>
<td>Responses here indicated whether or not the respondents would be good candidates for lower level supervision. Shared accounts remove individual accountability (and the ability to monitor individual behaviour characteristics) and as such the only profiling possible would be at the user class level.</td>
</tr>
</tbody>
</table>

Table 6.6: Rationale of Survey Questions
Chapter 6: Improving system security in healthcare

In addition, the following questions were originally included, but were then subsequently removed during interim revisions of the material (their purpose was to indicate whether a class of user had any potential for effecting a security breach requiring technical knowledge, but they were removed once it was established that the IT & Security Personnel audience was not to be surveyed):

- How would you rate your level of operating system knowledge?
- Please indicate your familiarity with the concepts of malicious software and system flaws.
- Are you involved in any software development work?

These were not considered relevant to general users given the level of system access that they are normally granted within Derriford (e.g. they are not involved in software development work and may not even be familiar with the concept of an Operating System).

6.6.4 Questionnaire Distribution

For the convenience of the staff at the Hospital, the survey was tied into another study that was being conducted at the same time as part of an undergraduate final year project (Holben 1995). This work was concerned with conducting a more detailed survey of attitudes towards security within the hospital and it was decided that the questions relating to class profiling would also provide useful background information in this context. The required questions were, therefore, incorporated into this study, with co-operation being seen as beneficial to all sides (in that the security attitudes survey also formed part of an internal...
study being conducted within Derriford). The full questionnaire contained 37 questions and a copy is provided in appendix A.

A total of 200 questionnaire forms were provided and distributed to staff by managers within the hospital, acting on behalf of the Trust Information Doctor. From the profiling viewpoint, it was hoped that at least ten samples would be obtained in each staff category, but this could not be guaranteed as the HCE controlled the distribution and staff cooperation could not be relied upon in all cases.

6.6.5 Results and general observations

In actual fact, the overall return was 74 responses (i.e. 37%), with all staff categories except "researcher" being represented. Whilst this was considered to be a very good response rate considering the size of the full questionnaire, the fact that the total number then had to be broken down into the sub-categories for each user class meant that in some cases the response base was still very small.

As a first stage of analysis, it was necessary to disregard the responses from any respondents who indicated that they did not use hospital information systems at all. The effect of this upon available responses within each staff category is shown in table 6.7.
Chapter 6: Improving system security in healthcare

<table>
<thead>
<tr>
<th>Staff Category</th>
<th>Total Responses</th>
<th>Responses where IT system used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consultant</td>
<td>11</td>
<td>7</td>
</tr>
<tr>
<td>Junior Doctor</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Nurse</td>
<td>23</td>
<td>18</td>
</tr>
<tr>
<td>Administrator</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>Secretary</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>PAM</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>Clerk</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Total</td>
<td>74</td>
<td>61</td>
</tr>
</tbody>
</table>

Table 6.7: Responses to profiling questionnaire

A full breakdown of the relevant responses is provided in table 6.8, with the columns consequently corresponding to basic class profiles that may be derived (the number of usable responses to each question are represented in the table by the values in brackets).

In most cases, the profile values are single figures representing tallies of the valid responses. However, there are a number of exceptions where entries are broken down into a number of sub-columns:

- typical hours of work (2 entries - earliest start and latest finish across responses);
- IT hours per day (3 entries - minimum, mean and maximum hours across responses);
- applications (3 entries - number of respondents who use them whole day, part day or less frequently);
- data (3 entries - number of respondents who create, access or update data).
<table>
<thead>
<tr>
<th>Staff Category</th>
<th>Consultant</th>
<th>Junior Doctor</th>
<th>Nurse</th>
<th>Administrator</th>
<th>Secretary</th>
<th>P.A.M</th>
<th>Clerk</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total usable responses</strong></td>
<td>7</td>
<td>3</td>
<td>18</td>
<td>11</td>
<td>5</td>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td><strong>Typical hours of work</strong></td>
<td>(3) 8.15</td>
<td>(1) 9.00</td>
<td>17.00</td>
<td>(11) Shifts (24hr)</td>
<td>(11) 8.30</td>
<td>(19.00</td>
<td>(4) 8.30</td>
</tr>
<tr>
<td><strong>IT hours per day</strong></td>
<td>(3) 1</td>
<td>1.6</td>
<td>3</td>
<td>(3) .5</td>
<td>1.3</td>
<td>3</td>
<td>(17) .2</td>
</tr>
<tr>
<td><strong>Systems</strong></td>
<td>(17) .2</td>
<td>(18)</td>
<td>(11)</td>
<td>(5)</td>
<td>(5)</td>
<td></td>
<td>(11)</td>
</tr>
<tr>
<td>Standalone PC</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>PC on a network</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Terminal</td>
<td>10</td>
<td>3</td>
<td>17</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Remote system</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Other</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>Applications</strong></td>
<td>(7)</td>
<td>(3)</td>
<td>(18)</td>
<td>(11)</td>
<td>(2)</td>
<td>(5)</td>
<td>(12)</td>
</tr>
<tr>
<td>PAS</td>
<td>1</td>
<td>2</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Clinical workstation</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Clinical Laboratory</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Radiology</td>
<td>2</td>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial Systems</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Theatres</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>(7)</td>
<td>(2)</td>
<td>(18)</td>
<td>(10)</td>
<td>(5)</td>
<td>(5)</td>
<td>(12)</td>
</tr>
<tr>
<td>Patient Care / Diagnosis</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>10</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>Patient Administration.</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Personnel</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Resource Management.</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>General HCE Administration.</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lab, Radiology or other service dept.</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td><strong>Access from &gt; 1 terminal ?</strong></td>
<td>3 (7)</td>
<td>3 (3)</td>
<td>8 (18)</td>
<td>6 (11)</td>
<td>1 (5)</td>
<td>3 (5)</td>
<td>10 (12)</td>
</tr>
<tr>
<td><strong>Located in different areas ?</strong></td>
<td>0 (3)</td>
<td>3 (3)</td>
<td>1 (8)</td>
<td>0 (6)</td>
<td>0 (1)</td>
<td>2 (3)</td>
<td>3 (10)</td>
</tr>
<tr>
<td><strong>Share group password ?</strong></td>
<td>3 (7)</td>
<td>0 (3)</td>
<td>2 (18)</td>
<td>1 (11)</td>
<td>0 (5)</td>
<td>2 (5)</td>
<td>3 (12)</td>
</tr>
</tbody>
</table>

Table 6.8: Healthcare “class” profiles
Some principal observations that can be made from the profiles are given below (remembering again that they are specifically linked to the Plymouth environment and cannot necessarily be extrapolated to represent the same classes of user in another domain).

- Typical hours of work would allow some sort of valid boundaries to be determined for all classes except Nurse, for whom irregular shift patterns appear to be the norm.

- All classes except for Administrator, Secretary and Clerk appear to be relatively consistent in terms of the number of hours per day spent using IT systems.

- Junior Doctors and Clerks appear likely to access systems from more than one terminal, with the opposite being the case for the class Secretary. Results were rather less conclusive for the other classes.

- The majority of classes access a wide variety of applications and data. However, some clear trends may be observed (e.g. Nurse and Clerk classes are more likely to use the Patient Administration System (PAS) than any other application. Nurses are most likely to utilise patient care or administration data).

- It is possible to classify the use of systems, applications and data as being frequent, occasional or never for each class of user - giving a broad profile of their relative IT usage.
Chapter 6: Improving system security in healthcare

- It appears that users in classes Consultant, Nurse, Administrator and Clerk are only likely to access IT systems from one area of the HCE.

- Group passwords seem rare in most cases (with the possible exceptions of Consultants and P.A.Ms, for whom insufficient samples were available to allow more definite conclusions to be drawn). As such, user-specific behaviour profiles would be feasible in most cases.

6.6.6 Limitations

It is possible to identify some notable weaknesses with the results and the underlying profiling strategy that was employed.

Firstly, careful examination of table 6.8 reveals some significant contradictions in the results. For example:

- whilst the responses indicated that Nurses might typically use IT systems for less than an hour a day (with a maximum claim of two hours), eight respondents later claimed that they accessed the PAS application for the “whole day”;

- whilst all secretaries indicated that they used IT systems, only two of them indicated that they actually used any applications (which in both cases was the PAS - an application that does not handle much of the data subsequently indicated in response to the follow-up question).
Whether such discrepancies resulted from simple misunderstandings or carelessness on the part of the respondents is unclear. However, whatever the cause, they serve to highlight a weakness of questionnaires for approaching this form of profiling.

Secondly, the rather small response rate in the majority of the staff categories means that the results cannot be regarded as a conclusive basis for profiles. They do, however, allow an indication of how the concept could be applied (although in practice, a more formal and thorough investigation would need to be conducted by the HCE involved).

Finally, the questions included only enabled rather basic information to be collected - again making it difficult to draw definite conclusions across the responses. More detailed investigation could potentially have yielded more descriptive and accurate profiles, determining for example:

- which systems are used for which applications?
- which applications are used for which data?

However, this level of analysis was considered too difficult to present in a questionnaire format (a view that appears justified by the apparent misunderstandings of even the basic material that was distributed).
6.6.7 Conclusions

This work has served to provide an illustration of how the class profiling concept could be applied in healthcare and shown that it is possible to draw some clear distinctions between user groups based upon their general use of IT systems.

However, the limitations cited above mean that the resulting profiles are probably not of sufficient accuracy to be of use in practice, due to the poor representation of some users and the rather basic questions utilised. As a result, it is concluded that the best approach for profile development may be to automate aspects of the data collection and analysis (e.g. using neural networks to observe and learn system usage patterns across user groups over an extended period).

As such, it is still envisaged that class level profiles could be used as the first stage of supervision in a practical scenario (alongside more comprehensive user-specific profiles).

6.7 A survey of existing Intrusion Detection Systems

The previous section has established various means by which IMS intrusion monitoring and supervision could be achieved. However, before proceeding to consider the actual system design, it is necessary to acknowledge a number of previous examples of systems in this area. This section presents an overview of earlier work and, in doing so, attempts to highlight which aspects of the previously identified functionality were encompassed and what, if any, weaknesses were apparent.
Chapter 6: Improving system security in healthcare

A recent survey of existing intrusion detection systems (Mukherjee et al. 1994) identified several examples of known approaches to the problem. These are listed below, along with other principal examples that were identified during the course of the research:

- ComputerWatch (Dowell and Ramstedt 1990);
- DIDS (Distributed Intrusion Detection System) (Snapp et al. 1991);
- Discovery (Tener 1986);
- Haystack (Smaha 1988);
- IDES (Intrusion Detection Expert System) (Lunt 1990);
- ISOA (Information Security Officer's Assistant) (Winkler 1990);
- MIDAS (Multics Intrusion Detection and Alerting System) (Sebring et al. 1988);
- NADIR (Network Anomaly Detection and Intrusion Reporter) (Hochberg et al. 1993);
- NIDX (Network Intrusion Detection eXpert system) (Bauer and Koblentz 1988);
- NSM (Network Security Monitor) (Heberlein et al. 1990);
- SecureNet II (Androutsopoulos et al. 1994);
- Wisdom and Sense (Leipins and Vaccaro 1989).

The significant number listed may immediately raise the question of why a new IMS would be needed. However, it should be noted that many of these systems have been developed to a prototype stage only and, in addition, several represent proprietary solutions used by specific organisations.
Whilst all of the systems share the common basic aim of “intrusion detection”, the approaches and capabilities vary significantly between the cases. Table 6.9 presents a high level summary of the capabilities, based upon details either explicitly stated in, or inferred from, the documented descriptions of each system. With regard to the table, a rather loose definition of “real-time” has been adopted and systems are judged to meet the criteria as long as data is analysed during the active session as soon as it becomes available (as opposed to being batched and then assessed en masse at some later time, e.g. the end of each day or week). Entries in the “attacks potentially detectable” column refer to the types of intrusion that each system is designed to combat (although there may be circumstances where a variety that is not indicated might still be trapped).

<table>
<thead>
<tr>
<th>System</th>
<th>Detection methods</th>
<th>Attacks potentially detectable</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Behaviour Profiles</td>
<td>Rules</td>
</tr>
<tr>
<td>ComputerWatch</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>DIDS</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Discovery</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Haystack</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>IDES</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ISOA</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MIDAS</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>NADIR</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>NIDX</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>NSM</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SecureNet II</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Wisdom &amp; Sense</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 6.9: Overview of existing intrusion detection systems

Of course, this table only provides summary indications of the capabilities of each system and there will obviously be differences between the systems even where they have a tick in
the same column. As such, a full assessment / evaluation would require that various further questions need to be asked (e.g. How are behaviour profiles comprised in each case? On what aspects are rules based? How is monitoring implemented? What are the detection strategies for each represented category of intruder? Are the systems application / platform specific?). In general, the reader is referred to the original references for answers to such questions, but a brief summary of the notable advantages and / or limitations in each system is presented below.

- **ComputerWatch** is principally an audit trail data reduction tool, which applies limited expert system rules to detect anomalous behaviour and simple security breaches. It is intended as a tool to assist the system administrator, minimising the amount of data that needs to be viewed. The system runs on the V/MLS operating system and aims to provide three levels of intrusion detection - system, group and user (although analysis of group events was described as a planned future enhancement in the referenced material).

- **DIDS** is actually an outgrowth of the NSM project (described below), and is designed to overcome some of the deficiencies in the earlier system. Intrusion detection capability is extended from the LAN environment to arbitrarily wider networks. The architecture involves the use of host and LAN monitors which report any “interesting” events to a centrally-located DIDS director where they are analysed using an expert system (hence, distributed monitoring and data reduction is combined with centralised analysis). The system uses the same intrusion detection algorithm as Haystack.
• *Discovery* was specifically developed to detect unauthorised accessing of the US credit agency TRW's credit database and, as such, only aims to detect abuses of this application (rather than the operating system as a whole). Genuine subscribers are assumed to have more consistent inquiry patterns than hackers and the system uses a self-learning capability to determine frequently occurring patterns in this data. These patterns are used to construct rather basic user behaviour profiles, which are then compared to daily activity data in order to detect variances. The system outputs a file of exception data (with reasons) for later analysis by investigators.

• *Haystack* is another audit trail data reduction tool and particularly targets the detection of abuse by authorised users. The system was designed for detection of misuse on US Air Force mainframe computers. The anomaly detection rules used are based on constraints imposed by official security policy and behaviour models for users and groups. Data is transferred from the mainframe to a PC for analysis (with a typical days activity being processed within a few hours). Intrusion detection is performed in three ways: monitoring *notable events* (single events that alter the security state of the system - e.g. access violations), *special monitoring* (involving the specific supervision of nominated users or objects) and *statistical analysis* (based upon the monitoring of a number of minor behaviour characteristics, such as the number of files created or pages printed, and looking for anomalies).
Chapter 6: Improving system security in healthcare

- **IDES** has been developed by Stanford Research Institute (SRI) and aims to provide a system-independent approach to the detection of security violations. The system is based upon a combination of statistical anomaly detection and expert system rules. A notable feature is that it is able to adaptively learn a subject's behaviour patterns, as profiles are updated from audit records on a daily basis. IDES was enhanced to develop a new prototype entitled NIDES (for Next-generation IDES), which utilises new statistical algorithms. It is better integrated and has an improved security officer interface over the original version.

- **ISOA** also incorporates statistical and expert system analysis methods. The system is designed for Unix-based workstations and monitors hosts as well as users. Two methods of anomaly detection are supported, termed *preliminary* (which occurs in real-time with the collection of audit data) and *secondary* (which is invoked at the end of a user session). A key point in this case is that the detail of supervision increases in response to the identification of initial anomaly indicators (these stages of monitoring are termed "concern levels"). ISOA behaviour profiles also include a historical record of previous supervision (e.g. how often a particular threshold has been violated in the past).

- **MIDAS**, as its full name suggests, is specifically designed for systems running the Multics operating system and is used on the US National Computer Security Center's Dockmaster mainframe. User and system level profiles are maintained and three types of rule are used to review audit data: *immediate attack* (based on the same principle as generic intrusion indicators), *user anomaly* and *system*
state (based upon departures from user and system profiles respectively). As with IDES, the system has the ability to update behaviour profiles, but in this case at the end of each user session. The system incorporates four tiers of rules and, as with ISOA, the firing of some rules will cause monitoring to advance to the next level.

- *NADIR* is a system designed for Los Alamos National Laboratories' Integrated Computing Network (ICN) and runs on a SUN SPARCstation II, collecting data from three service nodes on the network. The system supplements manual audit record review by comparing weekly network activity against expert rules and reporting suspicious behaviour to the system security officer (SSO). Tools are then provided to allow the SSO to perform follow-up investigations. The expert system rules are developed through audit analysis and consultation with security experts, with user's suspicion level being the sum of all rules that are triggered.

- *NIDX* was a prototype system developed by Bellcore specifically aimed at the monitoring of Unix systems. The rules in the knowledge-base model information about file system and user objects (classifying them as being either public or restricted for read, write and execute activities) and contains a set of policies and heuristics for detecting and assessing activities in relation to these. Although monitoring may have the potential to identify various classes of intrusion, the approach seems rather limited in terms of the factors considered. The user profiles are only based on a small number of historical usage statistics (e.g. login times, file and directory access, typical working directories and frequency of
command errors) and, as such, they do not attempt to fully characterise user behaviour. Similarly, only a limited number of indicators are monitored to detect Trojan Horse type activity (e.g. modification of file or directory permissions).

- **NSM**, developed by the University of California, differs from most of the previous cases in that it analyses LAN traffic rather than audit trail records. Behaviour profiles in this case are related to expected network traffic, which are compared to current traffic by a simple expert system. The NSM approach has several advantages. Firstly, it is capable of monitoring heterogeneous hosts by interpreting standard network protocols (e.g. TCP/IP). Secondly, LAN monitoring allows almost instant access to data (as opposed to waiting for it to be retrieved and transmitted from audit records). Lastly, the fact that NSM passively listens to the network logically protects it from subversion (e.g. by clandestine users). A limitation is that the system is restricted to LAN monitoring and may, therefore, be weak on detection of external penetrators who access via dial-up lines.

- **SecureNet II** is intended to deliver threat identification and recognition in Integrated Broadband Communication (IBC) networks. The system utilises both neural networks and rule-based expert systems approaches to interpret monitored behaviour information, and attempts to detect and predict user intentions from it. In the event of anomaly detection the system selects appropriate countermeasures and presents them as recommendations to the security officer.
Chapter 6: Improving system security in healthcare

- *Wisdom & Sense* is a statistically based anomaly detection system operating on a Unix system, but analysing data from VAX/VMS-based hosts. The most notable feature of the system in comparison to the others here is that it has the ability to derive its own rulebase of normal behaviour from historical audit data. These form a *rule forest* of human-readable rules, allowing them to be inspected, modified and supplemented by human experts. The rulebase may contain between $10^4$ and $10^6$ rules (of 6 to 8 bytes each), which can be searched in around 50ms.

From this review it can be seen that only a minority of the existing applications provide complete coverage of the full range of monitoring requirements and that none of the surveyed systems are specifically targeted at the healthcare environment. Those systems that simply present anomalies to the system administrator (e.g. ComputerWatch, Discovery) only provide a one stage improvement over conventional audit trails and, as such, may be considered inappropriate to the requirements originally identified. It is desirable that the detection of an anomaly should not require administrator intervention in all cases and the system should have some degree of autonomy (an administrator will still be a mandatory requirement of course). It should be noted that none of these systems appear to have considered the issues of system auditing and advanced initial user authentication or, indeed, the advantages of combining them within the intrusion monitoring framework.

With these points in mind, it is possible to identify the need for a new monitoring system that can more adequately address the comprehensive supervision requirements that exist in healthcare systems.
6.8 Conclusions

This chapter has served to illustrate the various categories of computer abuse to which healthcare systems may be exposed and has highlighted the applicability of real-time supervision techniques as a means of addressing the problem. To this end, a number of suitable monitoring techniques and behavioural characteristics have also been identified that would assist in the detection of all classes of intrusion, in both healthcare and other domains.

It is now necessary to move on from this to consider how these can be incorporated into a security system. This issue is covered in the next chapter, which presents a comprehensive conceptual design for an intrusion monitoring system.
CHAPTER 7

Conceptual design of an
Intrusion Monitoring System
Chapter 7: Conceptual design of an Intrusion Monitoring System

7.1 Introduction

This chapter describes in more detail how the elements identified in the previous sections would be integrated into the framework of a comprehensive monitoring system. Limited aspects of implementation will then be addressed in the chapters that follow.

A full IMS would be implemented using an expert system approach and would operate by comparing current system activity against information held in a knowledge base. The knowledge base would effectively maintain two "models" of activity for reference by IMS:

- normal activity (i.e. the user behavioural profiles);
- intrusive activity (i.e. the generic rules).

These models will determine what types of activities and events the system will look for and, as such, an event will be judged to be indicative of a suspected intrusion if:

- it is compatible with intrusive activity;

OR

- it is incompatible with normal activity.

7.2 Operating Cycle

In common with other similar systems (Andreoutsopoulos et al. 1994) the standard operating cycle of IMS (following the initial user identification and authentication) would involve the key phases listed below.
Chapter 7: Conceptual design of an Intrusion Monitoring System

- **Monitor**
  
  The principal function of the system, involving the collection of data relating to current user behaviour and system activity and the subsequent comparison against existing profiles and rules.

- **Detect**
  
  The identification of unusual behaviour or activity (indicated by departures from the profiles etc.).

- **Classify**
  
  The investigation of the detected anomaly. Does it represent an intrusion? If so, what type (e.g. is it caused by a user or a process; a penetrator or a misfeasor etc.)?

- **Respond**
  
  The automatic action (countermeasure) performed in response to the intrusion.

It should be noted that whilst they are logically separated here, the stages of monitoring, detection and classification may not be so distinct in terms of actual processing. For example, monitoring has the specific aim of detecting any anomalies, and the fact of detection assumes a knowledge of what to look for (and, hence, some degree of implicit classification). The operating cycle is illustrated in figure 7.1.
As the figure implies, the monitoring activity is envisaged as a continuous process that will be performed at all times whilst the host system is in operation. In a fully networked implementation (see section 7.7) this cycle would describe the Hosts relationship to each IMS Client process.

7.3 Response to suspected intrusions

The existence and operation of IMS should remain transparent to the user unless an anomaly is suspected. As stated above, a suspected intrusion will cause IMS to automatically perform some further action (the nature of which will vary depending upon the type of intrusion involved). Options here include:

- issuing of an explicit request (or challenge) for further authentication;
- recording of details in an intrusion log for later inspection / investigation;
- immediate notification of the system manager (i.e. an intrusion alarm);
- phased reduction of permitted behaviour (whereby less activities become possible as alert status increases);
Chapter 7: Conceptual design of an Intrusion Monitoring System

- locking of the intruder's terminal;
- termination (or suspension) of the anomalous session / process.

The degree of automatic response is an important consideration and, as indicated above, must be matched to the severity of the suspected intrusion. For example, if there is high confidence that an activity represents an intrusion or if a particularly serious breach is suspected, then the maximum countermeasure response should result. However, in lesser scenarios more limited responses will be appropriate (e.g. to the extent of just writing details to the intrusion log).

There is an obvious danger that any option which allows the user to continue working whilst the anomaly is investigated would also allow more time for an intruder to cause damage. At the other extreme it would be undesirable for the system to terminate a session or process without a very high degree of certainty that an intrusion was in progress (e.g. there are many scenarios in healthcare where such action could disrupt care delivery or threaten patient safety, and it is unlikely that HCPs would tolerate such an occurrence more than once). Therefore, the first two options above are considered to be the most appropriate as initial forms of response.

In practice, there are several possibilities for the type of challenge that the system could issue in the event of a suspected intrusion. The original system password would obviously be inadequate, given that it may have already been compromised in order for an intruder to have gained access in the first place. It is desirable that the challenge be such that it allows any legitimate user to resume work quickly with minimal interruption (i.e. it should be easy
for them to overcome, whilst still trapping impostors). A suggestion is that a (short) series of question and answer type challenges be posed to the user (Haga and Zviran 1991), who would then need to answer them correctly in order to proceed further. These could be based upon cognitive and/or associative information, with valid responses having been obtained and stored in conjunction with the original user profiling. If several (e.g. 5 to 10) such questions were to be obtained from users during profiling then the challenge could be based upon a random selection from the set (further reducing the chance of impostors being able to compromise the system). A diagrammatic example of this process is shown in figure 7.2.

![Diagram of IMS operation](image)

**Fig. 7.2 : Potential IMS operation**

There are, however, a number of scenarios in which this approach would be ineffective. Firstly, it must be remembered that any form of “authentication-based” challenge would be an inadequate countermeasure against misfeasors. They would obviously be able to
respond correctly to such challenges (having originally supplied the information themselves) and then continue with unauthorised activity. There is a solution here in the realisation that continuing anomalies would lead to a succession of intrusion alerts; an event which would be suspicious in itself. At this point, the IMS response could then change to a method which would effectively combat misfeasors as well (e.g. a session lock or a trigger for system manager investigation). Nevertheless, this would still enable misfeasors to continue for longer than other classes of intruder (albeit with intermediate challenge(s)) before the system locks them out.

A second problem / exception relates to suspected malicious processes - these cannot be issued with a challenge to which they may respond and verify their legitimacy. This in turn places more importance on the correctness of the resulting IMS response (e.g. the dangers of suspending / deleting a legitimate, and possibly essential, process or failing to take positive action against a genuinely destructive one).

Finally, some classes of anomaly (for example, login failures based on unrecognised user identities) cannot be tied to a specific user and, as such, the issue of a challenge based on profile information is again inappropriate. However, it is conceivable that some form of generic challenge could be issued (the answer to which would be known by legitimate system users), with invalid responses causing the IMS to proceed to its next level of countermeasure (e.g. system manager notification, terminal lockout).

In conclusion, therefore, any anomalies deemed most likely to represent one of these scenarios should be addressed using some response other than the automatic challenge.
7.4 IMS Architecture

At a high level, the IMS architecture is based upon the concept of a centralised Host handling the monitoring and supervision of one or more Clients running on local workstations. The purpose of the Clients is to collect the required data relating to user and process activity and respond to any suspected intrusions detected by the Host.

All behaviour profiles, generic rules and such like are maintained securely at the Host, which also handles all of the analysis and the main bulk of other processing associated with the supervision. By contrast, the Client involves no local data storage and acts almost exclusively as an agent of the Host.

At a lower level, the Host and Client systems will be comprised of a number of modules, each handling a different aspect of the overall intrusion monitoring task, as illustrated in figure 7.3. The modules shown are intended to represent the conceptual elements of the system, but could also equate to the coded functional elements in a full implementation. The key aspects of this design are defined in the sections that follow.
Chapter 7: Conceptual design of an Intrusion Monitoring System

Fig. 7.3: IMS Architecture
7.4.1 Anomaly Detector

The purpose of the *Anomaly Detector* will be to analyse user and process activity for signs
of suspected intrusion, comparing it against the behaviour profiles (class and user-specific)
that apply to the current users (claimed) identity as well as against the generic intrusion
rules. It is envisaged that this will be comprised of a number of further sub-modules, each
handling a specific aspect of anomaly detection (e.g. keystroke analysis).

Various aspects of *Anomaly Detector* functionality will now be discussed.

7.4.1.1 Maintenance of an intrusion alert status

The detector will maintain an *alert status table*, with entries existing throughout the life of
each user-initiated session or process to indicate the level of detected anomalies and thereby
the confidence of a potential intrusion. Each entry will be in the basic form shown in figure
7.4 and will be examined and updated each time activity data relating to the relevant user /
process is analysed.

<table>
<thead>
<tr>
<th>User / Process ID</th>
<th>Alert Status level</th>
<th>Idle time</th>
<th># previous challenges</th>
<th>Session start</th>
<th># access violations</th>
</tr>
</thead>
</table>

*Fig. 7.4 : Structure of Alert Status table entry*

It is envisaged that, at its most basic, the "alert status level" could be a simple aggregate
value based on the number of behavioural anomalies detected and intrusion rules satisfied
(with the monitored characteristics and rules having been weighted to indicate their significance). The entry relating to "idle time" will be used to allow the phased reduction of the alert status level after certain periods of inactivity. Recording a tally of "previous challenges" would then be used as a safeguard to determine whether the level of IMS response should be escalated in response to an anomaly even if the alert status is currently low (i.e. as a result of the phased reduction). As the figure illustrates, the table might also be used to store other information, such as the time of session / process initiation or the number of access violations incurred. These would be used to for the purposes of on-going comparison against behaviour profiles (for example, session start time could be used to derive the current session length) and would also be required to be maintained throughout the live of the session. It should be noted that some of the table entries are most applicable in the context of monitoring user sessions and will be redundant in the case of process supervision.

The alert status level would increase in response to a number of conditions:

- departures from user-specific behaviour profile;
- departures from user class profile;
- satisfaction of generic intrusion indicators.

However, given that the class profile only represents behaviour in the most general terms, it is conjectured that even total departure should only be able to take the alert status to a certain level (e.g. enough to flag the user for attention, but not to cause a challenge).
Under normal circumstances the detector would commence supervision of a session with an alert status of zero (i.e. no suspicion of an intrusion). However, factors such as failed login attempts, system configuration anomalies and the like could cause it to begin with a non-zero status so that it is essentially more sensitive to further anomalies in the initial instance.

The alert status would be reduced after successful challenges or after a sufficient period of normal activity to allow the system to discount the previous anomaly.

7.4.1.2 Restriction of user activities

It is considered feasible for the alert status level to be interlinked with the types of activity that a subject is allowed to perform, such that a phased reduction of permitted behaviour would occur as the level increases (as suggested in section 7.3). In this way, highly sensitive activities and/or information could be denied if there is any doubt over the current users legitimacy, whilst still allowing more mundane activities to continue. The approach would demand that a maximum alert status threshold be associated with each of the activities or objects which IMS is to control. If the current status level was then to exceed this, the activity or object would become unavailable.

For example, consider the thresholds in table 7.1 associated with two objects (wordprocessor and patient database) and the activities create and delete file. If the current alert status level was 5 then the user would not be permitted to access the patient	
database or to perform any file deletion. However, the creation of a file using the wordprocessor application would still be possible.

<table>
<thead>
<tr>
<th>Activity / Object</th>
<th>Alert Status Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>wordprocessor</td>
<td>8</td>
</tr>
<tr>
<td>patient database</td>
<td>2</td>
</tr>
<tr>
<td>create file</td>
<td>8</td>
</tr>
<tr>
<td>delete file</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 7.1: Alert status threshold table

Such a threshold table would be maintained within IMS, but the values would initially need to be assigned (and, if necessary, subsequently updated) by the system administrator.

It must be said that the potential for error would make this approach inappropriate in many healthcare scenarios (for example, the denial of data access in a direct care application could be most unwelcome). In any case, it would be advisable for the system administrator to be notified whenever behaviour restrictions were being imposed so that the situation could be investigated (in case legitimate users were being unintentionally impeded).

7.4.1.3 Suspension of supervision

In some cases it is envisaged that continuous behaviour monitoring at all times throughout a user session may not be strictly necessary or even advantageous. This is especially true in the case of the mechanisms aimed solely at the detection of penetrators (e.g. keystroke analysis). The rationale here is that, after a reasonable amount of uninterrupted behaviour
analysis (i.e. with no challenges and no significant periods of user inactivity), the monitoring system should have been able to accurately determine the legitimacy of the current user. If an impostor is not suspected at this point then it is extremely unlikely that further monitoring will detect one (indeed, monitoring for longer than is necessary would simply allow more opportunity for false rejections to occur and place an additional load on the system).

In view of this, it is considered that monitoring activity during the following periods is likely to be most crucial in terms of impostor detection (with supervision being temporarily suspended at other times):

- during the time immediately after the start of the session (when the authenticity of the user has yet to be conclusively proven);

- during the time after any significant periods of inactivity (during which an impostor could potentially have replaced the legitimate user).

Important considerations here would obviously be the period of monitoring necessary before suspension of supervision and also what length of time would constitute the "significant period of inactivity" necessary for it to be resumed. Suggested periods would be up to 5 minutes of activity before suspension (in order to allow a sufficient appraisal of the user to be made), followed by 2-3 minutes of inactivity as a trigger for monitoring to resume (as this length of time could have allowed sufficient opportunity for impostor
intervention). However, in a practical implementation both of these aspects could be configurable so that the optimum levels could be established.

It should be noted that this approach would not be adequate for detection of misfeasor activity, as this could very well proceed after authentication has been established. Therefore, if suspension of monitoring was still to be incorporated, it would be sensible to periodically reintroduce supervision at random intervals as an additional safeguard (this would also help to guard against a situation where an impostor / penetrator might be able to replace the authorised user without there being a significant period of inactivity - e.g. coercion of the legitimate subject).

This idea is primarily suggested as a means of minimising the likelihood of false rejections in the practical context. However, a further advantage in the context of practical implementation would be that it would reduce the significant processing overhead that would be associated with continuous monitoring in an environment with several Client machines.

7.4.2 Profile Refiner

In a full implementation it would be desirable for IMS to utilise user activity data in two ways - to analyse for anomaly detection and as the basis for updating behaviour profiles. This second point recognises the possibility that user behaviour may legitimately alter over time (e.g. as a result of access to new applications, improvements in typing ability etc.). The purpose of the Profile Refiner would, therefore, be to provide an automatic means for user-specific profiles to be updated to account for such changes.
It would be most appropriate for the Profile Refiner to be based upon a Neural Network approach (Fausett 1994), given that the inherent ability to analyse and recognise patterns could allow behavioural characteristics to be identified that might not be apparent to a human observer. In this way, the effectiveness of the system would have the potential to improve over time, in that it could gradually learn more patterns of legitimate activity for each user (building upon the foundation provided by the generic rules and the initial profiles). It might also be possible to determine which of the profiled characteristics provide the best discriminators for each user and thereby establish (for example) primary, secondary and tertiary level behaviour indicators (with the primary level representing the most reliable identity verifiers). This hierarchy could also be extended to allow for the fact that some characteristics may represent negative indicators (i.e. those that, despite refinement, are found to cause a high level of false alarms).

It would be undesirable for the Profile Refiner to utilise data that is later found to be anomalous. Refinement should, therefore, only take place after the termination of user sessions / processes (provided, of course, that no intrusions were proven during this time). However, it is also considered sensible to allow refinement to proceed if any challenges that were generated were correctly answered by the user (the reason being that the generation of the alert may be indicative that legitimate behaviour has departed from the profile and that refinement is, therefore, necessary). However, in order to help guard against the recognised problem that misfeasors will answer challenges correctly, refinement should be performed on the proviso that the number of alerts raised was small relative to the length of the session (i.e. 2 alerts in a 3 hour session would be acceptable, whereas the same number
in a 10 minute session would be very suspicious). Additionally, any activity occurring
during periods where supervision of the relevant aspect was suspended could not reliably be
used for profile refinement.

User-specific profile records would also incorporate a series of flags to indicate whether the
individual behaviour characteristics are ready to be used in supervision or still being
developed. This will allow a gradual training period to be defined for new user profiles
without the IMS continually generating intrusion alerts (the flags would also allow a specific
"refinement only" period to be established for existing profiles that have proved to be
inadequate for the legitimate user). The purpose of associating flags with each profile
characteristic is so that some degree of monitoring could still continue whilst other aspects
are being (re)trained. The flags could also be used to allow the total disablement of some
aspects of monitoring if, for example, some characteristics are found to be inappropriate to
certain users.

Data relating to process activity would not be used for refinement as the generic rulebase
would remain static (unless specific information on new intrusion methods is introduced by
the system administrator).

### 7.4.3 Recorder

The Recorder handles the short-term storage of system activity data during the period of a
user session. Upon termination the information will be picked up and used by the Profile
Refiner, provided that the session was not considered anomalous.
7.4.4 Archiver

The Archiver will collect data relating to all system activity and store it in a long-term archive (in the same manner as a traditional audit trail), providing a more permanent record of activities and suspected anomalies.

The storage will occur regardless of whether sessions / processes are regarded as anomalous and details of all security relevant events will be archived. Such events will include login failures, intrusion alerts, authentication challenges, suspended sessions and the like.

The basic format of the archive records would be as shown in figure 7.5.

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>User / Process ID</th>
<th>Logged Event</th>
<th>Privileges utilised</th>
<th>Resources utilised</th>
</tr>
</thead>
</table>

Fig. 7.5 : IMS Archive record structure

However, in order to conserve storage space, it may be desirable in some scenarios to only record details of certain types of event. The Archiver should therefore be configurable to suit the preferences of the establishment involved (note that the same would not necessarily be true for the Recorder as this would always need to collect information on any activities for which profile refinement may later occur).

The long-term retention period of archived details would be determined by the security policy of the HCE involved.
Chapter 7: Conceptual design of an Intrusion Monitoring System

7.4.5 Collector

The Collector represents the interface between the IMS and the existing information system / applications, with the responsibility for obtaining information on all relevant user and system activity.

The module would be required to operate in such a way as to encompass, but be independent of, all system applications. It is envisaged that this could be best achieved by implementation at the operating system (OS) level, such that key events also lead to IMS notification. For example, a significant proportion of data collection could be based around the interception and redirection of selected OS interrupts and service requests (such as file input / output, application execution, keyboard input). These would be monitored with two objectives:

- to collect data on those events which pertain to monitored behaviour characteristics;

- to identify those events which may affect the security of the system (for comparison against generic intrusion indicators).

In some cases the required data could be obtained directly from audit trail records on the underlying system (as with some of the other systems discussed earlier, such as ISOA and Wisdom & Sense). However, with certain aspects (e.g. keystroke analysis) the required information will not be maintained in audit trails and implementation may, therefore, require
a significant number of operating system links. Whilst this would serve to make this aspect of IMS very system specific, it would be considerably more efficient than attempting to modify each individual application to specifically provide relevant information to IMS via an API (Application Program Interface) or similar method. The system specific coding of the Collector would only need to be done once, whereas modifications would be required to all current and future applications (which would be likely to be a non-trivial undertaking and potentially impossible in the case of commercial packages where source code may be unobtainable).

The resolution of data collection would be determined at the Host by the System Administrator.

7.4.6 Responder

This module resides in the Client and handles the task of responding to anomalies detected by the Host. The operation of the Responder would centre around the continuous monitoring of the alert status transmitted by the Host, with increases in the level triggering appropriate actions. The nature of response might include issue of a user authentication challenge, suspension of a session or cancellation of a process (as previously identified in section 7.3).

In some implementation scenarios, the Responder might also be responsible for handling the initial user identification and authentication process that is required to gain access to the system in the first instance.
7.4.7 Communicator

The Communicator provides the network communications interface between the Host and the Client(s) operating on the local systems. As such, the functionality of this module is duplicated on both sides of the link.

The principal functions would include transmitting user and process information to the Host and then subsequently keeping the Client(s) informed of the current alert status.

If implemented in a heterogeneous environment, the Client side of the module would be responsible for resolving any operating system differences that exist within the monitoring domain so that information could be presented to the Host in a consistent, standardised format.

7.4.8 Controller

This module is provided for use by the System Administrator to allow the operation of the IMS system to be configured.

On the Host side, configuration would apply to the following modules:

- *Anomaly Detector*, e.g. behaviour characteristics to consider / prioritise, generic rules in operation;
- *Profile Refiner*, e.g. frequency of refinement, acceptable thresholds for challenges within a session;
• Archiver, e.g. level of detail required, specific events to record or exclude from logging.

On the Client side, the operation of the following modules could be controlled:

• Collector, e.g. the level of data collection (linked to the characteristics being monitored by the Anomaly Detector).

• Responder, e.g. the level of response required at each alert status level.

These settings would obviously be controlled and recorded through the Host system. The configuration of the local Client(s) would then be established at the time of session initiation.

In addition, several other features would also be provided under the auspices of the Controller module. These would include facilities such as user profile management, update of the generic rulebase and the like.

7.4.9 Profiles

As previously identified in sections 6.5.2 and 6.5.3, IMS profiles could conceivably hold a range of identification, authentication and behavioural information relating to legitimate users.

The profiles would use a number of methods to represent measures of user behaviour:
Chapter 7: Conceptual design of an Intrusion Monitoring System

- frequency tables (e.g. for file access);
- means and standard deviations (e.g. for keystroke / typing profiles);
- ranges (e.g. valid access times);
- lists (e.g. for valid access locations).
- a combination of methods (e.g. a list of valid access locations which also indicate the relative frequency of use).

The profile data obviously requires secure storage to prevent unauthorised browsing or tampering by potential impostors. If users were able to modify profile information it would be possible for them to adjust the records of other users to match their own (and therefore allow them to access the account in place of the legitimate owner). Whilst disclosure of the profile statistics may not initially appear to pose such a threat, it could still be a problem in the case of a determined impostor. For example, if the characteristics of the "target" user were known, the impostor would have a concrete statement of what he / she would be required to mimic. An alternative option would, of course, be to subsequently enlist the help of an accomplice with a comparable profile.

At the very least, this dictates a requirement for encrypted storage, as used with the password files in the majority of commercial operating systems (Gait 1978; Morris and Thompson 1978). However, the proposed method for storing profiles, and other security management information, goes beyond this and will be discussed in section 7.7.3.
Chapter 7: Conceptual design of an Intrusion Monitoring System

7.5 IMS Implementation

The IMS concept is considered most appropriate to implementation in a networked environment, for the following reasons:

- standalone systems will most often be dedicated to a single user. As such, more traditional authentication and access controls (e.g. passwords) will probably be sufficient to ensure security if they are correctly implemented.

- implementation of a full IMS would be likely to degrade the performance of a standalone system.

- networked systems provide more potential for collecting monitoring information. Many statistics (e.g. access location, resource usage) would not be appropriate to a standalone environment.

The sections that follow will consider the options for IMS implementation in different types of networked environment, with a view to establishing which approach would be best to pursue for the purpose of a demonstrator system.

7.5.1 PC implementation

In this scenario the Host would be centralised (on a dedicated machine) with multiple IMS "Clients" being used to monitor activity on the individual workstations. The purpose of the Clients would be to collect any activity data that is generated locally (e.g. keystroke
timings) and to enforce IMS restrictions in suspected intrusion scenarios (e.g. issue a challenge, lock access to the system etc.).

In such a scenario it would be necessary to maintain the security of the IMS Clients on the individual machines to ensure that their operation cannot be compromised (e.g. by a malicious user trying to avoid detection).

Fig. 7.6: IMS in networked PC environment

7.5.2 Minicomputer implementation

There are two alternative strategies for implementing IMS in a minicomputer environment:

- complete implementation as process(es) running on the main machine;
- host implementation on a physically separate processor / system, taking information from a client on the central host.

The first scenario has the disadvantage that IMS would be consuming processing resources of the system and possibly degrading other applications as a result. In addition, the IMS process(es) may be vulnerable to interference from clandestine users or other (malicious) processes.

The second method partially overcomes these issues as IMS processing would be limited to the Client process(es), but has the disadvantage that separate hardware would be dedicated to IMS operation (as in the PC network solution). However, this is considered the better option and is illustrated in figure 7.7.

Fig. 7.7: IMS in minicomputer environment
Additional problems may arise in either scenario from the fact that minicomputer systems will often rely upon "dumb" terminal devices. In these cases a limitation is placed upon the information that IMS can collect locally. For example, it would not be possible to obtain the measurements required to perform keystroke analysis (although the addition of some kind of hardware "black box" may overcome this - at a price).

As indicated by the earlier discussion in chapter 3, both types of system configuration are significantly represented in HCEs, so in that sense either approach to IMS would serve a useful purpose. However, as the previously identified intrusion detection prototypes were all based in minicomputer environments, it was considered more important from the research perspective to pursue an implementation in the rather unsupported PC environment. The discussion of this aspect will begin in chapter 8.

7.6 Advantages and disadvantages of the IMS approach

This section presents an overview of the principal advantages and potential disadvantages that are perceived with the IMS approach, with particular reference to the use of the concept in healthcare systems.

7.6.1 Advantages

The advantages of the IMS approach were largely established as part of the original justification for investigating the concept. However, for completeness, the main points are also listed here.
Chapter 7: Conceptual design of an Intrusion Monitoring System

- **Improved security**

  This is advantageous in any information system, and is achieved here due to the continuous nature of supervision. User authentication is no longer restricted to the discrete judgement(s) possible with passwords and misuse will be identifiable a lot earlier than with traditional auditing. In addition, the fact that much of the supervision is based upon behavioural characteristics makes it more difficult for users themselves to undermine security (e.g. by allowing colleagues unauthorised access to their accounts) as they cannot easily transfer these abilities to other users.

- **Cost**

  Advantages here result from the fact that it is possible to implement the concept entirely in software at the user end, whereas many frequently suggested authentication enhancement schemes (e.g. Smart cards, other biometric methods) would be reliant upon specialised equipment at each user workstation. This makes the technique particularly suited to financially constrained environments such as healthcare.

- **Convenience**

  This comes from the fact that the supervision can be performed transparently, in a non-intrusive manner. In addition, the fact that the IMS would demand nothing special from the users (e.g. they are not required to remember additional password-type information or possess any physical token) means that its
operation should not contradict or undermine the existing staff culture in any way.

It should be noted that these advantages specifically address some of the special constraints that normally apply in the healthcare environment.

7.6.2 Disadvantages

There are a number of inherent disadvantages / weaknesses in the concept of IMS (and any other type of comprehensive monitoring and supervision system). The principal concerns are highlighted below.

- The operation of IMS Clients and / or data collection will consume system resources and may degrade overall performance. The collection of detailed audit trail data typically degrades machine performance by between 5 and 20 percent (Wolfe 1992; Mukherjee et al. 1994). An IMS performing full behavioural monitoring and testing of generic intrusion rules would be envisaged to introduce a similar burden.

- Transmission of data from Clients to the Host will result in a loss of network bandwidth and a loss of timeliness of data. However, this factor is also shared with most of the other intrusion detection systems previously identified in chapter 6 and, in any case, data would still be available and analysed far faster than with traditional auditing.
Chapter 7: Conceptual design of an Intrusion Monitoring System

- Maintenance of the IMS itself would entail a more significant management/administration burden in the affected host systems. For example, correcting problems with behaviour profiles would be a more complex operation than cancelling a forgotten password. At the same time, however, other duties (such as inspection of audit trails) would be reduced, so the new demands would at least be somewhat offset.

- The overall concept of continuous supervision raises a question of user acceptance. It is conceivable that there may be mistrust and resentment of the system on the grounds of it being seen as a means of monitoring legitimate work and staff performance as opposed to just guarding against intruders. It would, therefore, be important to ensure that the system is perceived as a "Caring Mother" rather than a "Big Brother". This issue is discussed in somewhat more detail in chapter 8.

In general terms the likely advantages when compared to other means of protection are considered sufficient to outweigh these points in the healthcare context. However, the most significant of the concerns here would be the potential effects on speed/performance of existing applications (given the high availability requirements in many cases). It is not really possible to give a definite statement on the extent of the problem as this would depend upon the practical implementation of the IMS, as well as the general processing power and operational overheads (e.g. number of concurrent users) of specific host systems.
7.7 Intrusion monitoring in a wider security architecture

The concept of the IMS as discussed to this point has been presented as an independent system in its own right. However, healthcare security needs go beyond this and it is therefore sensible to expect that an intrusion monitoring facility would be required as part of a wider, more comprehensive, security system framework.

This section describes how this objective may be achieved by examining IMS in the context of a Comprehensive Integrated Security System (CISS), which has been defined in a previous research programme (Shepherd et al. 1990; Patel and Sanders 1991; Muftic et al. 1993). As the concept is fully documented in these reference sources, this section only provides a brief overview of CISS itself and concentrates upon how IMS could be incorporated within the framework.

7.7.1 The Comprehensive Integrated Security System

The basic intention of CISS is to provide a single, flexible system that can be integrated into any application as required (i.e. as an overlaid service with the entire system sitting between end-users and host applications in much the same way as is intended with IMS), allowing all activities to be transparently mediated by the security system. Service provision is based upon a limited number of well-defined procedures / techniques. In this way each can be analysed and tested so that certain standards of security can be "guaranteed" to the user.

The architecture supports the full range of Open Systems Interconnection (OSI) security recommendations (ISO 1989), which include the following principal functions (it can be seen that the majority of these are outside of its intended scope of IMS):
Chapter 7: Conceptual design of an Intrusion Monitoring System

- invocation;
- identification and authentication;
- key generation and key distribution;
- encryption and decryption;
- digital signature;
- verification.

The CISS architecture is based upon the interaction of ten autonomous agents. These represent the logical components of the system, with each implementing one, or a group of, strictly defined functions. Separate agents exist to handle the management and coordination of security services, the establishment of secure associations within and between domains, and user authentication and monitoring, as well as providing facilities for system administration and fault recovery. The specific functionality and logical relationships between the ten agents is described in detail in the referenced material, but for summary purposes their names are listed below:

1. User Agent (UA);
2. Security Administrator Agent (SAA);
3. Security Services Agent (SSA);
4. Security Mechanisms Agent (SMA);
5. SMIB Agent (SMIBA);
6. Agent for Operational Environment Interactions (OPENA);
7. Association Agent (AA);
Chapter 7: Conceptual design of an Intrusion Monitoring System

8. Inter-Domain Communications Agent (IDCA);
9. Monitoring Agent (MA);
10. Recovery Agent (RA).

A further important aspect of the architecture is the Security Management Information Base (SMIB) which acts as the repository for all data relevant to the security system (i.e. users, security functions and system objects).

The implementation of CISS is based upon the concept of security domains, which refer to sets of users, applications networks and systems that are tied together by a common security policy. Examples of domain composition could, therefore, be different departmental LANs within the same HCE or even different HCEs within the same local authority (the compatibility of security policies being made possible through the adherence to a common standard; for example, an enhanced set of the baseline security guidelines presented in chapter 4, which also consider encryption issues).

Each domain would include one or more Security Management Centres (SMCs) to handle and control all local security activities - analogous to an extended version of the IMS Host. These will perform the majority of management functions and will securely hold information relating to user authentication, access rights and other supervisory characteristics. SMCs could be operated by the host HCEs to avoid problems of trust, with inter-domain activities then being mediated by the SMCs in the domains involved.
Chapter 7: Conceptual design of an Intrusion Monitoring System

7.7.2 Intrusion monitoring in the CISS framework

The integration of the IMS concepts into the CISS framework is initially simplified by the fact that the ideas of intrusion detection and user supervision have both been considered as potential aspects of the CISS architecture. However, neither have been defined in any detail and, as such, the integration of the IMS concept provides a means to rectify this.

At the most basic level, the concepts previously presented in the discussion of IMS can be seen to integrate into the CISS architecture by providing the functionality required by the Monitoring Agent (MA). The role of this agent is to monitor system usage for audit trailing and fault reporting purposes - which approximates to the general idea behind IMS.

However, it is worth examining the existing definition of the MA in more detail to establish a more specific relationship. The main functions that can be ascertained are as follows:

- continuous monitoring of all security relevant events;
- access and management of an (encrypted) security log;
- event handling (i.e. responding to anomalies).

Further definition, taken directly from the previously cited work by Shepherd et al, states that the agent:

"... could be an AI-based module that will detect problems and even likely problems before they occur, and take the necessary actions for preventative or remedial measures."
Both of these elements of the specification continue to relate closely to the IMS concept. The existing documentation also suggests that some connections may be drawn between IMS and the Recovery Agent, in so far as the latter is defined as having responsibility for:

- all security violation detection;
- CISS error recovery.

However, the majority of the further discussion of the RA in the existing material tends to concentrate on the second aspect, leading to the conclusion that this agent is more concerned with handling CISS internal errors than those resulting from anomalous user/process behaviour. As such, it is possibly better to regard IMS-type "security violation detection" as being within the terms of the MA through the "continuous monitoring" and "event handling" aspects.

Given that only this broad specification exists, it is definitely feasible for the intrusion monitoring and recovery aspects of CISS to be implemented in the manner suggested by IMS without introducing major conflicts with either of the existing designs. It is, therefore, possible to consider in more detail how the integration would be achieved.

7.7.3 Distribution of IMS functionality within the CISS

It has already been established that the role of the SMC is analogous to that of an extended IMS Host and, therefore, the latter would be encompassed at this level. By examining the separate functional elements that comprise the IMS design, it can be seen that it is actually the Anomaly Detector that would equate most closely to the basic concept of the MA.
Chapter 7: Conceptual design of an Intrusion Monitoring System

However, the MA would actually handle more than just this aspect and a more comprehensive breakdown of the relationships of IMS modules to CISS agents is given in table 7.2.

<table>
<thead>
<tr>
<th>IMS Module</th>
<th>CISS Agent(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anomaly Detector</td>
<td>Monitoring Agent</td>
</tr>
<tr>
<td>Archiver</td>
<td>Monitoring Agent (via SMIB Agent)</td>
</tr>
<tr>
<td>Collector</td>
<td>Operational Environment Interactions Agent</td>
</tr>
<tr>
<td></td>
<td>User Agent</td>
</tr>
<tr>
<td>Communicator</td>
<td>User Agent</td>
</tr>
<tr>
<td>Controller</td>
<td>System Administrator Agent</td>
</tr>
<tr>
<td>Profile Refiner</td>
<td>Monitoring Agent (via SMIB Agent)</td>
</tr>
<tr>
<td>Recorder</td>
<td>Monitoring Agent</td>
</tr>
<tr>
<td>Responder</td>
<td>User Agent</td>
</tr>
<tr>
<td></td>
<td>Monitoring Agent</td>
</tr>
</tbody>
</table>

Table 7.2: Relationship of IMS modules to CISS agents

All aspects of IMS stored data would be absorbed into the CISS framework as part of the SMIB. The storage of data in the SMIB is logically separated into six conceptual segments, as follows:

1. an identification segment for entities and other network resources;
2. an extended security segment holding security profiles of users, programs and other network resources;
3. a secure associations segment which maintains details of active network users and their associations (e.g. membership of closed groups, secure teleconferences etc.);
Chapter 7: Conceptual design of an Intrusion Monitoring System

4. an extended access control segment holding the specification of access control parameters for network users and resources;

5. a security log - for recording of security relevant information and use in recovering after system failures;

6. a confidential segment for active entities.

It is envisaged that the integration of the IMS data groups would affect the majority of these segments in some way and the likely relationships are shown in table 7.3 below.

<table>
<thead>
<tr>
<th>IMS data group</th>
<th>SMIB Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-specific behaviour profiles</td>
<td>Extended Security</td>
</tr>
<tr>
<td>Class level behaviour profiles</td>
<td>Extended Security</td>
</tr>
<tr>
<td>Generic rulebase</td>
<td>Extended Security</td>
</tr>
<tr>
<td>Alert Status table</td>
<td>Secure Associations</td>
</tr>
<tr>
<td>Alert Status thresholds for object access</td>
<td>Extended Access Control</td>
</tr>
<tr>
<td>Archive</td>
<td>Security Log</td>
</tr>
</tbody>
</table>

Table 7.3: IMS data storage in the SMIB

Whilst this is fine at the conceptual level, it is necessary to consider this issue in slightly more detail to see how data would actually be stored. Looking first at where the behaviour profiles would be incorporated, it is clear that they could be integrated as an expansion of the existing data on user entities. This is confirmed by the description from Shepherd et al (1990), which recognises the use of behavioural data and briefly mentions the opportunities that would be offered by a future, “semi-intelligent” version of the system that was able to take account of “users’ habits”. However, the current SMIB data structure for system users is somewhat vague, containing only the following “fields”:

219
Incorporation of user behavioural data would require a data structure significantly more complex than this, with the further information basically representing an expansion of the "extra data" field. At the very least, this would involve the inclusion of the field structures shown in figure 7.8.

![Diagram of System User Entity data]

**Fig. 7.8 : Extension of System User Entity data**

The representation of behavioural characteristics would involve the use of a variety of underlying data structures, as appropriate for each specific measure involved (for example, keystroke characteristics could be stored using a series of digraph matrices). Note that the status flag associated with each behaviour characteristic would be used to indicate whether it is enabled, disabled or being (re)trained in relation to the particular user entity.
As a result of the further "secret" data within the structure, it would be desirable to increase the use of encryption beyond just the user password to also include the challenge data and some elements of the behaviour data (i.e. those aspects that could potentially be mimicked if details were made known to impostors).

The maintenance of the generic rulebase does not fit neatly into the existing SMIB structure and would require an additional structure type. For each intrusion rule it would be necessary to store the series of events against which the MA would be required to monitor in order to identify anomalies, as well as (an indication of) the necessary remedial action that should be taken in the event of detection (e.g. alert status increase, immediate issue of a user challenge etc.). For example:

<table>
<thead>
<tr>
<th>Rule 1</th>
<th>Event Conditions</th>
<th>Required Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule N</td>
<td>Event Conditions</td>
<td>Required Actions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

User and system supervision facilities will then become available as part of the range of common security services offered to applications by CISS. Examples of specific services (of both discrete and continuous nature) would be as follows:

- **Discrete services**:
  - enhanced authentication;
  - anti-virus protection;
  - system audit.
Chapter 7: Conceptual design of an Intrusion Monitoring System

- Continuous services:
  - real-time user supervision (at class and individual levels);
  - process monitoring and control;
  - anti-virus protection;

However, it can be deduced from this list that IMS is more oriented towards providing services to system/application entities than to end-users. Therefore, the Security Administrator would be likely to be the only "user entity" with the ability to selectively enable/disable the services that are provided.

The IMS Profile Refiner would utilise an existing inter-relationship between the Monitoring and SMIB Agents as a means for the automatic update of behaviour profiles in the SMIB.

Other CISS agents will handle further aspects of IMS functionality which would otherwise need to have been provided in an IMS-specific fashion. For example, the IMS sub-system would utilise the generic facilities of the System Administrator Agent to allow management and administration of user profiles, available supervision features, generic rules and such like.

As indicated at the beginning of this section, the formulation of the CISS framework was the basis of a previous research project. A further, independent research programme is currently in progress that should lead to the realisation of certain elements of the
architecture in a practical context (Rhodes 1995). It is envisaged that an IMS-type architecture could consequently be incorporated at some later time.

7.8 Conclusions

The IMS concept is not intended as a total replacement for conventional authentication and access control methods (although in some cases it will offer an opportunity for more dated approaches to be replaced). In the majority of systems, supervision could be incorporated alongside other methods to complement the security already provided.

In addition, it will have little or no effect upon the need for physical security and personnel-related measures within an organisation. There are also some important aspects of "logical" security that are not addressed (e.g. protection of data communications) which highlight the potential need for a wider framework such as that of CISS.

Finally, the advocated monitoring approach would incur too great a processing overhead if attempts were made to implement it on a scale above that of a LAN. As such, further protection is required to safeguard WAN activities and inter-HCE interactions. This issue is explored in more depth in the penultimate chapter of the thesis.

Before this, however, the next two chapters present a yet lower level examination of (certain aspects of) the IMS concept. This is based around the practical investigation and evaluation of a specific real-time supervision technique that is considered appropriate for (although again not limited to) use in healthcare systems. Following an experimental study,
it is then shown how the technique may be incorporated into a framework similar to that which has been presented in this chapter.
CHAPTER 8

Real-time supervision using Keystroke Analysis
8.1 Introduction

Having established the basic design options and required functionality for the IMS, it was also necessary to evaluate aspects of supervision in a more practical context. However, rather than present a general overview of several methods it was decided that a detailed treatment of a specific technique would be more appropriate.

The earlier discussion has identified that passwords provide an unreliable basis for user authentication and that stronger methods are necessary, using techniques that are more difficult, if not impossible, to forge. One of the necessary requirements of IMS will be the ability to perform continuous authentication to ensure that a valid user is present at all times during a session.

Previous work has identified three main categories of information that may be used to validate a user (Wood 1977):

- something the user knows (e.g. a password);
- something the user has (e.g. a token such as a card or key);
- something the user is (e.g. a biometric such as fingerprint or voice pattern).

In general, the last option appears to be the strongest, as possession may not be easily transferred to other people in the same way as secret knowledge or a token. This point was recognised by the 1990 AIM Working Conference on Data Protection and Confidentiality (Commission of European Communities 1991b), which cited a need in healthcare for
Chapter 8: Real-time Supervision using Keystroke Analysis

“identification possibilities which are directly connected with the individual staff member”. Biometric measures were specifically mentioned as being worthy of further investigation. However, as previously mentioned, the cost of the technology required to successfully implement most biometric methods would largely preclude its uptake in healthcare. What is, therefore, required is a biometric measurement that can be obtained without requiring any form of additional hardware. Fortunately, such a characteristic can potentially be identified in the form of users typing style (or keyboard rhythm) and this has been selected as the basis for more detailed investigation.

The basic premise of the approach is that typing characteristics will be reasonably unique to each user, revealing an individual "signature" (analogous to those that can be identified with normal handwriting (Fairhurst et al. 1994)). This theory is lent weight by previous studies relating to the actual process of typing (Cooper 1983; Shaffer 1970). The concept of using keystrokes to assess identity was originally proposed by Spillane (1975) and can be claimed to provide a "behavioural" biometric measurement, in that the act of typing represents how a user does something as opposed to being a physiological characteristic.

Evaluation of the technique for inclusion in the IMS was considered to be appropriate as none of the intrusion detection systems surveyed in section 6.6 had incorporated keystroke analysis in their supervision strategies (although previous studies have been conducted independently, as will be discussed later).

In addition, it was considered that it would be easier to develop a user profile for keystroke characteristics than with most other behavioural aspects. A usable typing profile can be
constructed relatively quickly by having users partake in a specifically arranged profiling session. This would not be possible with any of the other characteristics discussed (e.g. analysis of operating system command usage, typical access times) as the only way to obtain an accurate measure of normal user behaviour in these cases would be by monitoring genuine, operational use of systems over a long period of time. Keystroke analysis was, therefore, considered more convenient as it would place relatively little demand on test subject availability (making it possible to use people who would not otherwise have been accessible for profiling). Conversely, the investigation of other methods would have required sustained access to test subjects which, if possible at all, would have considerably reduced the size of the test group in comparison to that which was eventually used in this study.

It was envisaged from the outset of the work that it would be possible to implement the technique transparently on the monitored systems and using entirely software-based methods. These factors would help to satisfy the convenience and financial constraints previously identified in the healthcare environment. However, it was also realised that this form of monitoring would only be appropriate for identifying the penetrator classes of intruder (as the keystroke signature alone could not be used to determine whether legitimate users are acting abnormally or abusing their privilege).

8.2 Keystroke Analysis Concepts

In order to provide a foundation for further discussion it is necessary to establish some of the theoretical concepts behind keystroke analysis. This includes consideration of how the
activity of typing may be used to identify and discriminate between users, as well as alternative implementation strategies. These issues are discussed in the sections that follow.

8.2.1 Typing characteristics

Before considering a monitoring strategy, it is necessary to identify suitable typing characteristics that may be used as a basis for analysis (and hence authentication).

The list below presents a series of factors that may be considered as a means of determining the keystroke "signature" of an individual user (note that some characteristics have more obvious potential than others).

- **Inter-keystroke times**
  The elapsed time between user keystrokes. These can be analysed in two ways, to generate *general* and *digraph-specific* measures. The general measure relates to the mean inter-keystroke time across *all* keystrokes, giving an indication of the users overall typing speed. The digraph-specific measure is more detailed and refers to the intervals between specific character pairs (e.g. the time between releasing "T" and pressing "H"). This measurement may potentially reveal details of the users characteristic *typing rhythm* and should be the more useful / accurate of the two inter-keystroke measurements.

- **Keystroke duration times**
  The length of time for which keys are held down. This can again provide general and key-specific measures.
• Typing error frequency

A basic measure of the user’s typing accuracy. Typing errors can be classified in a number of ways (Cooper 1983), but in simple terms such a measure could be determined by monitoring the usage of the delete key(s).

• Force of keystrokes

Whilst this would be likely to be quite distinctive in some cases (e.g. in differentiating between users who traditionally use manual typewriters and those that type entirely on computers), it is unfortunately impossible to obtain such measurements from the standard electrical keyboards found on most PCs and terminals.

• Keystrokes / Words per minute

The standard measures of typing ability. These would be likely to portray similar information to a general keystroke interval measure.

8.2.2 Categories of typist

Users are likely to differ dramatically in terms of typing styles and abilities, depending upon factors such as their familiarity with the keyboard, experience and any formal tuition.

Previous research (Card et al. 1980) has broken the classifications down into six categories based upon the average inter-keystroke time of the subject. These categories are shown in table 8.1.
Table 8.1: Typist Skill Categorisation

<table>
<thead>
<tr>
<th>Category</th>
<th>Average Keystroke Interval (Seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>0.08</td>
</tr>
<tr>
<td>Good</td>
<td>0.12</td>
</tr>
<tr>
<td>Average (Skilled)</td>
<td>0.20</td>
</tr>
<tr>
<td>Average (Non-Skilled)</td>
<td>0.28</td>
</tr>
<tr>
<td>Poor</td>
<td>0.48</td>
</tr>
<tr>
<td>Unfamiliar with keyboard</td>
<td>1.20</td>
</tr>
</tbody>
</table>

In theory it should be relatively easy to differentiate between users from different categories. Separation within a category may, however, be more problematic (for example, professional typists who underwent the same training may all exhibit a similar style).

8.2.3 Authentication strategies

As previously identified in table 6.3, there are two general approaches by which the concept of keystroke analysis may be incorporated into an authentication / supervision system. For the discussion purposes these will be referred to as the Static Identifier and Dynamic Identifier approaches.

8.2.3.1 Static Identifier

In this scenario authentication is based upon the entry of a static text string. This could be used in conjunction with normal login where a username and password are entered as usual, but rather than just authenticating the user from this information alone the system also analyses the way in which it was typed, providing a further level of authentication.
Chapter 8: Real-time Supervision using Keystroke Analysis

There are two fundamental factors that may strengthen this approach:

- the system analyses a constant text pattern for authentication rather than any arbitrary input. The samples used to generate user “typing profiles” can, therefore, concentrate on the specific text used for the identifier;

- the users typing is likely to be more consistent (automatic) for a well-known, regularly typed string than with other arbitrary input.

This combination theoretically means that quite a high degree of accuracy should be possible in the authentication.

The majority of reported studies and experiments to date appear to have concentrated in this area (Bleha et al. 1990; Joyce and Gupta 1990; Obaidat and Macchiarolo 1992; de Ru and Eloff 1995), with errors as low as 2% in the rejection of valid users.

8.2.3.2 Dynamic Identifier

Using this approach authentication is based upon any arbitrary text input, allowing greater scope for supervision in real-time during user sessions. Monitoring could occur either continuously or at selected (random) periods during the user session (the choice is likely to depend upon the processing overhead incurred by the monitoring system). This would provide the desired transparent (i.e. non-intrusive) means of authentication that is not currently possible with most other methods (even those based on biometric features).
In this scenario security is no longer reliant on a single authentication period but becomes continuous throughout the session. One advantage here is that it should serve to prevent "logical piggybacking", whereby an intruder attempts to utilise an unattended terminal that is already logged into another account. Whilst other methods exist to prevent use in this manner (e.g. physical keyboard locks), they tend to require positive action on the part of users if they are to be effective.

It is desirable that the decision period required to accept or reject a subject is as small as possible. The results of a previous experiment in this area (Leggett et al. 1991) cited that many impostors were detected in less than 100 keystrokes. Whilst this is impressive, it should be appreciated that this would still be a sufficient "window" in which to wreak havoc in some cases.

8.2.4 Evaluation of effectiveness

In order to determine whether keystroke analysis would be a worthwhile element of the IMS, it is necessary to obtain some measure of its effectiveness. As with other authentication systems based on biometric measurements (Cope 1990), this may be judged on the basis of two factors: False Acceptance Rate (FAR) and False Rejection Rate (FRR).

The false acceptance rate, which may also be referred to as the Impostor Pass Rate (IPR), relates to errors where impostors are falsely believed to be legitimate users. Conversely, the false rejection rate, which may also be termed the False Alarm Rate (FAR), refers to errors where the system falsely identifies the legitimate user as an impostor. These rates share a
mutually exclusive relationship, such that configuring the tolerance settings of a system to give good results for one will generally cause a degradation of the other (and, as such, it is not possible to attain optimum levels for both measures). This is illustrated in figure 8.1 below.

Fig. 8.1: Relationship between FAR and FRR errors

The "equal error" scenario is not really an appropriate compromise and a decision must, therefore, be made as to which rate should receive priority. In actual fact, the priorities will vary depending upon whether a static or dynamic authentication system is used.

- **False Acceptance Rate**

  This is obviously required to be as low as possible, otherwise impostors could remain undetected and the way is left open for breaches of security to occur. As highlighted by Joyce and Gupta (1990), in the case of a static verifier a FAR of
even 5% would be unacceptable as it would suggest that one in twenty unauthorised access attempts would be likely to succeed. If continuous authentication was used however, such a figure may be more tolerable as there would be increased likelihood of the system challenging the impostor at some stage during the session.

- **False Rejection Rate**

  From a security standpoint this statistic is of less concern, as false rejections do not facilitate security breaches. However, for the legitimate user the issue is obviously more significant. Again considering an error rate of 5% as the basis for comment, in the case of a static verifier the figure would probably be deemed acceptable as it would simply represent occasional inconvenience to legitimate users (one in twenty logins would fail wrongly - probably a lesser proportion than would normally fail simply as a result of typing mistakes anyway). However, for continuous authentication this frequency would be much more noticeable and, as indicated on the graph, could consequently be irritating to the user if the supervision system was to persistently disrupt the session to demand further authentication. Frequent instances of false rejection for specific individuals should trigger resampling their typing characteristics to create a more accurate profile.

In summary, therefore, minimising the FAR is the prime concern with static authentication, whereas a minimal FRR is of greater significance in the dynamic scenario.
A further important consideration (in the dynamic scenario) is the speed with which the identity assessment can be provided by the system (i.e. how many keystrokes could an intruder enter before being noticed?).

8.3 Implementation considerations for a Keystroke Analysis System

The effectiveness of the keystroke analysis concept was tested in practice using an experimental system developed as part of the research programme. The following sections present details of how this system was implemented, as well as the main findings observed during a large scale study.

Whilst a significant number of previous research efforts have been concentrated in this area, the majority have only considered the use of keystroke analysis in static authentication systems. As a consequence, this investigation concerned itself with the dynamic identifier approach, which was in any case considered most appropriate to the continuous supervision requirements of IMS.

In terms of the objectives of the investigation it was considered that, given the previous discussion of false acceptance and false rejection in section 8.2.4, the minimisation of the FRR should be a priority. The rationale here was that if the system was to be recommended as a viable supervision technique in HCE systems, then it should have the properties of transparency and convenience for legitimate users (which would not be the case with frequent false rejections).
It was further conjectured that if a FRR of 0% could be achieved, then what would effectively be observed in the results would be a “worst case” level for false acceptances (giving a good indication of the security that would be provided and the contribution that the technique would make to an overall IMS user profile). As such, this was considered to be the most suitable approach and was, therefore, the method pursued in the investigation (the means by which the minimal FRR was ensured is discussed in section 8.3.4).

A similar approach was previously used in a static verification study (Brown and Rogers 1993), where techniques were used to purposefully bias the system in favour of 0% false acceptances (bearing in mind that these are the key consideration in the static context). Whilst establishing a “worst case” FRR would have served no advantage in our investigation, the fact that such a study has been performed does provide further justification for an experiment that pursues the converse goal in the dynamic scenario.

The chosen strategy is also significantly different from that presented in the only previously available study of dynamic keystroke analysis (Leggett et al. 1991). Here the authors attempted to establish reasonable minimums for both FAR and FRR errors - an approach which was previously discounted as an inappropriate compromise in section 8.2.4. (note: the results yielded in this case were a FRR of 11% and a FAR of 13%).

It was considered that an experimental evaluation could be conducted using a system as shown in figure 8.2, with the typing characteristics of a series of test subjects being profiled and each then subsequently submitting further text samples for testing. The profiles and
samples would then be compared off-line by a monitoring system to determine impostor
detection effectiveness (note: each test sample would be used as an impostor attempt by
being run against everyone else’s profile).

Fig. 8.2 : Keystroke analysis experiment overview

However, prior to staging a full investigation, a series of preliminary tests and a pilot study
were conducted to answer some general questions about how the system should operate.
The principal considerations here included:

- which typing characteristics would be most suitable for inclusion;
- which keystrokes should be included in the analysis;
- how to obtain an adequate typing profile;
- how to perform authentication.

The following sections detail how these issues were addressed.
8.3.1 Typing characteristics evaluated

From the various typing characteristics described in section 8.2.1, the following were considered to be the most likely discriminators and were selected for evaluation in an initial pilot study using 13 test subjects:

- Inter-keystroke time;
- Typing error frequency;
- Keystroke duration.

Even prior to the pilot study it was anticipated that the analysis of inter-keystroke times for specific digraphs would be the best discriminator, with the other measures serving to provide supplementary information to strengthen the process. The tests confirmed this view, with significant differences being observed between valid subjects and impostors in terms of the proportion of valid interval timings that each generated. However, the general keystroke interval test was found to be less robust, with a FAR of 66%.

Typing error frequency was found to be quite variable for certain subjects whilst others appeared to be relatively consistent. This indicated that the measure would not be an adequate discriminator in its own right, but it was nevertheless used in the tests as an additional consideration to the inter-keystroke timings. However, it was actually found to be even worse than the general inter-keystroke test, yielding a FAR of 72%.

The strength of the keystroke duration measurements was considered even more doubtful and was dropped following some preliminary tests conducted before the pilot study. The
reason for this was that all test subjects, regardless of their typing skills, appeared to
generate very similar results. The general conclusion drawn from this was that the sampling
"window" for key depressions is too narrow to obtain any information from which to
distinguish between the majority of users. Only users with a very "deliberate" keystroke
style may be an exception to this. Although attempts were made to strengthen the test, no
noticeable improvements resulted and the keystroke duration measurements were, therefore,
dropped entirely.

8.3.2 Keystrokes selected for analysis

It was decided that the system should not attempt to gather information for all user
keystrokes, as only a subset would be likely to exhibit the distinctive rhythm necessary for
authentication.

Monitored keystrokes were, therefore, restricted to alphabetic characters and spaces.
Numeric characters were not included, although it is acknowledged that in some contexts
(e.g. financial systems) the monitoring of numeric input could be potentially characteristic
(and, indeed, that alphabetic input may not feature so strongly). Punctuation and other
"special" characters were excluded as their use is generally less frequent.

No differentiation is made between the use of upper and lower case characters. Whilst
some previous experiments (Leggett and Williams 1988; Umphress and Williams 1985)
have chosen to exclude upper case input from analysis (presumably on the grounds that they
occur with less frequency within normal prose) this disregards the fact that in some contexts
(e.g. certain command line environments) upper case input may occur with more frequency.
The only scenario in which the character case is likely to affect timing is when a character is generated in conjunction with the *shift* key. However, the frequency of *shifted* input is not considered sufficient to significantly distort results and with this system the decision was therefore taken not to differentiate.

Previous research (Leggett and Williams 1988) has also attempted to differentiate between digraphs according to which half of the keyboard they appear on and, hence, the hand(s) used to type them (i.e. left or right hand only, left to right or right to left) to determine whether any further discriminating power could be gained. However, this additional filtering resulted in FARs and FRRs that were noticeably higher than when all digraphs were treated equally and, therefore, no such tests were performed in this investigation (a further, independent observation is that the distinction between left and right hand digraphs is only sensible when talking in terms of reasonably skilled typists. Inexperienced typists may be less consistent in terms of which hand they use and may, for example, type a character on the left side of the keyboard with their right hand).

Deleted keystrokes are not used in the creation of inter-keystroke time profiles. It is anticipated that if the second character of a digraph was entered as a result of a mis-stroke then the inter-keystroke time observed would be unlikely to be representative of a user's normal typing style. Therefore, the interval times associated with keystrokes that are subsequently deleted should be disregarded.

It is to be expected that users may pause during typing (e.g. due to distraction, stopping to think or read ahead etc.) resulting in inter-keystroke times that are uncharacteristic of their
normal typing rhythm. Use of these times could distort both keystroke profiling and subsequent monitoring and, therefore, inter-keystroke times exceeding a certain threshold should be ignored. For this reason a highpass filtering level of 750ms was incorporated into the system (chosen to be compatible with Card's classification of a user who is unfamiliar with the keyboard layout and pauses to look for keys). However, there is a danger that, upon discovering the existence of the highpass filter in the monitoring system, an impostor could attempt to fool the system by typing deliberately slowly. As such, an additional safeguard must be provided to detect unnaturally slow typing.

8.3.3 Creation of Keystroke Profiles

It is necessary to obtain a reference sample of the typing characteristics of each legitimate user for use as the basis for future authentication (i.e. a keystroke profile). It is obvious that this profile will be extremely important in determining the accuracy / effectiveness of the resulting keystroke analysis.

The considerations that must be addressed at this stage are the selection of an appropriate reference text upon which to base the profiling and issues of how best the profile timings may be processed once obtained.

8.3.3.1 Selection of a Reference Text

The main considerations in the selection of the reference text are sample size and composition.
The main issues regarding sample size are that the profile text must be sufficiently long enough to ensure that the following requirements can be satisfied.

- Individual digraphs can be sampled sufficiently to allow typical performance ranges to be established. In theory, the more samples that are used to create the mean and standard deviation values, then the more effective the profile should be. Conversely, profile entries based on a very small number of samples will be unreliable indicators of typical performance. Therefore, the experimental system only utilises profile entries if five or more digraph samples were used to create them. This attempts to ensure that "weaker" profile entries are ignored, whilst still retaining a significant number of usable digraphs.

- The users "natural" typing style emerges. Too small a sample may not accurately reflect the users normal typing style. For example the user may try to type as fast as possible to complete the task and / or may make more typing mistakes than usual (e.g. the user may feel pressured due to the knowledge of being monitored). However, a sample of adequate size should ensure that the users normal style is forthcoming.

- The requirements for test composition can be satisfied (see below).

The performance of legitimate users was experimentally examined using profiles generated using text samples of varying sizes (a 541 character passage of text was used as the basis for this and was sampled two, four and eight times to generate the different profiles). The
effectiveness of each profile was then gauged by observing the percentage of incompatible inter-keystroke times in a subsequent sample, as shown in table 8.2.

<table>
<thead>
<tr>
<th>Total Sample Size (chars)</th>
<th>% keystrokes not matching profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>1082</td>
<td>50</td>
</tr>
<tr>
<td>2164</td>
<td>41</td>
</tr>
<tr>
<td>4328</td>
<td>35</td>
</tr>
</tbody>
</table>

Table 8.2: Effect of profile sample size

From this it can be seen that sample size makes a significant difference to the results. Authentication noticeably improves using the profiles based on larger text samples and hence the large text sample appears to be warranted.

However, it is unlikely that increasing the sample size further would be beneficial given that the second doubling of the sample size did not yield as dramatic a reward as the first. Additionally, factors such as user fatigue (and patience) had to be considered during profiling and an even larger sample would have been likely to be unwelcome (note: in these tests a good typist was able to complete the 4328 character sample in 20 minutes, whereas a poor typist required around 45). From the administration perspective, too large a sample will mean the process of explicitly profiling each user will become burdensome in a large system. This point was made in another previous study (Joyce and Gupta 1990), which was generally critical of the use of large sample texts (it should, however, be noted that this experiment was concerned with the static identifier approach, where such significant samples were not required).
It was, therefore, considered that an overall sample size of between 4000 and 4500 characters was most suitable in order to construct a representative profile of typing.

The issue of sample composition refers to the number of different characters and character digraphs within the text. It would be almost impossible to generate a (readable) reference text that would incorporate all possible valid character digraphs. It is therefore to be expected that during subsequent monitoring some digraphs will be encountered for which no profile sample was taken. In such cases the system cannot make any judgement and must ignore the test data, which in turn reduces the opportunities for authenticating the input and hence the effectiveness of the system. As a result, it is important to minimise the frequency with which such cases will arise by selecting a representative text for initial profiling.

A statistical analysis typical English language text reveals significant differences in the frequencies with which particular characters (and hence character digraphs) occur (Beker and Piper 1982). It was, therefore, considered that user profiles would be likely to be more accurate if the reference text was representative of these frequencies. Further investigation of this possibility resulted in the discovery of a suitable text (with a length of 2202 characters) in which the character frequencies corresponded very closely to the expected frequencies in normal English (with the 30 most frequent character digraphs all being significantly represented). As such, acquiring two samples of this text was considered sufficient to satisfy both size and composition requirements. Tests revealed that with two samples, between 60% and 70% of the individual digraphs would be sampled the required
five times or more (the actual percentage depended on the number of digraphs excluded due to typing errors), thus providing a reasonably sound basis for later supervision.

It should be noted that the selection of the texts used as the basis for the test samples was not subject to the same strict conditions that applied to the reference text. The only requirement was that they were of an adequate length to provide a basis for monitor assessment. Issues of text composition were ignored based on the premise that monitoring should aim to be equally effective with any arbitrary text.

8.3.3.2 Filtering of profile sample timings

It was previously established that profiling filtered out deleted keystrokes and any inter-keystroke timings exceeding 750ms.

In addition, analysis of the keystroke profiles from the aforementioned pilot study revealed that many contained anomalous entries where the profiled digraph means were less than the associated standard deviations. Two potential explanations could be offered for this:

- one or more “rogue” keystroke times occurred during profiling that were artificially high or low (when compared to the rest of the values) and distorted the overall result (most likely in the case of experienced typists);

- the users performance for the particular digraph was too erratic for any “typical” keystroke pattern to be identified (the more likely explanation for poor or novice typists).
Irrespective of the cause, it was considered that the discrepancy would render the values themselves unsuitable for use in subsequent monitoring comparisons. Whilst the frequency of occurrence was shown to vary between test subjects, the number of digraphs available for use in monitoring was significantly reduced in some cases.

An experiment was, therefore, conducted to determine the extent to which each of the above explanations contributed to the problem and whether it would be worthwhile to introduce further filtering into the profiling module. The graphs in figure 8.3 illustrate eight digraphs for which the problem occurred within an experimental profile. The individual keystroke times that were observed in each case have been plotted in ascending order (note that interval times exceeding 750ms are not plotted as they would have been filtered out before calculation of the anomalous mean and standard deviation values).

It can be seen from the graphs that filtering out the highest and lowest sample times would overcome the problem in many cases (e.g. digraphs OF, AI, SO, EY and L_).

On this basis further tests were conducted to determine the level of filtering that should be applied. It was discovered that mandatory filtering of all digraph samples was unsatisfactory as samples that were perfectly valid were ignored, resulting in a weaker profile. Therefore, filtering was only applied if the initial calculation of mean and standard deviation was found to be anomalous.
Fig. 8.3: Digraph times where Standard Deviation exceeds Mean (inter-keystroke times against digraph occurrences)
It was considered that filtering should only be applied if ten or more digraph samples had been obtained (to ensure that a reasonable number of samples would remain after filtering upon which to base the profile entry).

Initial tests attempted to filter only the highest and lowest 5% of times, but this was found to be unsatisfactory where only a relatively small number of digraph samples had been collected. With the filter level increased to 10% the process was more effective, resolving standard deviation anomalies in approximately two thirds of the cases (with the remaining cases being ones where the profile entries were based upon less than ten samples and therefore filtering had not been applied anyway).

It was, therefore, considered that as the inclusion of filtering yielded a noticeable improvement in the number of usable digraphs (which should in turn result in more effective and reliable profiles for the affected users) it should be incorporated into the Profiler for the full study.

8.3.4 Authentication assessment

With the exclusion of the keystroke duration, general inter-keystroke time and typing error frequency measures, authentication is based entirely around statistics associated with the analysis of digraph-specific inter-keystroke times.

The user profiles store the mean and standard deviation of inter-keystroke times for each profiled digraph. These values are used to define the range of inter-keystroke times that are
considered "valid" for each digraph. It is expected that most keystroke times from legitimate users will be within this range whilst impostor keystrokes fall outside.

Various valid ranges were tested (including 0.5, 1, 1.5 and 2 standard deviations from the mean) and performance was observed in relation to legitimate subjects and impostors. The most acceptable range was found to be mean plus or minus 1.5 standard deviations. Statistics relating to invalid times (i.e. those falling outside the range) are used as the main basis for user authentication.

Three conditions trigger the issue of a challenge:

1. if the percentage of invalid timings (in the 100 most recent keystrokes) exceeds a subject-specific threshold;

2. if the number of consecutive invalid keystrokes exceeds a second subject-specific threshold;

3. if the proportion of inter-keystroke times exceeding the highpass filter level is greater than 50% (i.e. insufficient valid data for analysis). This is again based upon data from the 100 most recent keystrokes. The test was included to provide the safeguard against deliberately slow typing that was mentioned in 8.3.2.
All tests were considered to be of equal importance and it was necessary for subjects to pass all of them in order to be authenticated (however, the third method was not expected to affect the experimental study as the test subjects would not be actively attempting to beat the system in this way).

With dynamic data analysis, authentication judgements are made in real-time with a revised judgement after each user keystroke. By considering only the 100 most recent keystrokes in its calculations the monitor can ensure that its analysis is always based on up-to-date information. As such, it would be responsive to changes such as the legitimate user being replaced by an impostor.

It was realised that, even though the profiles were based upon significant typing samples, user performance in practice was unlikely to be totally compatible with them at all times. As a result, some degree of "invalid time" judgements must be tolerated for legitimate users. For example, a previous study (Leggett and Williams 1988) discovered that up to 40% of inter-keystroke times generated by a legitimate user could still be incompatible with their typing profiles.

The pilot study had used this 40% threshold to represent the percentage of invalid keystrokes at which an impostor alert would be generated. However, it was discovered that whilst all users did indeed generate a percentage of invalid keystrokes, the 40% threshold was rather excessive in most cases and simply served to allow leeway for impostors (leading to several false acceptances). At the same time, it was observed that in general, the percentage of invalid keystrokes generated by impostors was noticeably higher than that
which would be generated by the associated legitimate subject. This is illustrated in figure 8.4 where, for example, subject 12 typically generates only 20% invalid keystrokes against her own profile, whereas impostors average 35%. A threshold level of (say) 25% would, therefore, seem appropriate for authentication of this subject. However, the same level would be totally inappropriate for subject 1 (with 38% invalid digraphs against his own profile).

![Graph showing typical performance of impostors compared to legitimate subjects](image)

**Fig. 8.4**: Typical performance of impostors compared to legitimate subjects

On this basis, an appropriate solution was considered to be the use of subject-specific authentication thresholds, achieved by measuring legitimate subject performance against their own profile using subsequent typing samples. From this, the peak values observed for the percentage of invalid keystrokes and the number of consecutive invalid keystrokes were obtained and used as the basis for that subject’s future authentication thresholds. This was
the method by which the investigation was able to ensure that no false rejection errors occurred.

8.4 The experimental Keystroke Analysis System

This section describes details of how the experimental system was implemented, considering the computing platform chosen for the experiments and the components of the system itself.

8.4.1 Implementation environment

The experimental system was written in the C language and implemented in an IBM PC environment running under the standard MS-DOS operating system.

Keystroke timing data was collected by means of two PC hardware interrupts: 08h (timer) and 09h (keyboard). These are described in detail by Williams (1990), but an overview is given here along with an explanation of how they were utilised in the keystroke analysis system.

The timer interrupt is generated for each “tick” of the PC’s internal clock (the 8253 Programmable Interval Timer). This normally runs at 18.2 ticks per second, but can actually handle speeds of up to 3 MHz. As such, the keystroke analysis system increases this to 1000 ticks per second allowing a timing resolution of one millisecond accuracy.

The keyboard interrupt is generated twice for every user keystroke: once on key depression and again on key release. In each case a scancode is generated by the keyboard to tell the interrupt which key was involved (note that a scancode is normally a one byte value and is
completely distinct from any ASCII character code that the key may generate). In actual fact, each key has two associated scancodes - a make code and a break code - with the latter always being the equivalent of the make code plus 128. The replacement keyboard interrupt can therefore determine whether a key has been pressed or released simply by testing whether the scancode value is less than 128 (and if pressed, an inter-keystroke time can be stored). A final notable point regarding scancodes is that some keys (e.g. function and cursor keys) generate two byte codes, of which the first byte is always E0 hex. The replacement routine ignores these and does not store associated inter-keystroke times.

By using a combination of the timer and keyboard interrupts and testing scancode values it is, therefore, possible to time the interval between a key release and the next key press. This procedure is illustrated by the pseudo-code in table 8.3.

<table>
<thead>
<tr>
<th>Interrupt</th>
<th>Pseudo-code</th>
</tr>
</thead>
<tbody>
<tr>
<td>timer</td>
<td>(triggered every millisecond)</td>
</tr>
<tr>
<td></td>
<td>counter = counter + 1;</td>
</tr>
<tr>
<td></td>
<td>CALL old timer interrupt</td>
</tr>
<tr>
<td>keyboard</td>
<td>(triggered every time key pressed or released)</td>
</tr>
<tr>
<td></td>
<td>IF first iteration</td>
</tr>
<tr>
<td></td>
<td>counter = 0;</td>
</tr>
<tr>
<td></td>
<td>READ current keyboard scancode;</td>
</tr>
<tr>
<td></td>
<td>IF not extended key sequence</td>
</tr>
<tr>
<td></td>
<td>IF scancode = a &quot;make&quot; code</td>
</tr>
<tr>
<td></td>
<td>inter-keystroke time = counter;</td>
</tr>
<tr>
<td></td>
<td>counter = 0;</td>
</tr>
<tr>
<td></td>
<td>CALL old keyboard interrupt</td>
</tr>
</tbody>
</table>

Table 8.3: Pseudo-code for replacement interrupt routines
It can be seen that the time counter is incremented every time the timer interrupt is run and then reset every time the keyboard interrupt is executed. As such, the counter value accumulated between a key "break" iteration and a subsequent "make" iteration will represent the inter-keystroke time (in milliseconds). The time is stored, with the non-interrupt-driven code then determining the characters involved and calculating mean and standard deviation values for alphabetic / space digraphs.

It should be noted that the use of these interrupts serves to make the system specific to the PC architecture.

**8.4.2 System Modules**

In common with the overview diagram previously shown in figure 8.2, the experimental system was comprised of three principal modules, as will be described in the sections that follow. The implementation was very much geared to providing a platform for evaluation of the keystroke analysis technique and did not (at this stage) provide a fully functional security system.

**8.4.2.1 Profiler**

This accepts the initial text sample that is used to generate the keystroke profiles for legitimate users. The user enters a number of samples of the current "reference text", with inter-keystroke times being collected and used to generate the typing profile. The profile is then stored, along with the user name, for subsequent use by the Monitor module.
The Profiler presents the reference text in a small (three line) window that scrolls through the text in response to user input. The text currently being input is shown on the middle line and highlighted to distinguish it from the others (and enable subjects to more easily track their positions). User input occurs in a similar window below. If typing errors are made, the colour of the input text changes (white to red) from the point of the error onwards, so that users are able to easily identify where they went wrong. This is illustrated in figure 8.5 below:

![Profiler System](image)

Fig. 8.5: The Profiler system

A report file is created after profiling which includes an assessment of the subjects typing skill (according to the Card classifications), the total number of distinct digraphs sampled and the proportion of profile entries that are unusable (based upon the number of digraphs sampled less than five times and cases where the profiled standard deviation exceeds the mean).
8.4.2.2 Sampler

This module accepts further text samples from test subjects and stores all keystrokes and their associated timing data to a file for later use. These test samples are then used by Monitor to determine the effectiveness of the system. Figure 8.6 shows an example of sample collection in operation:

![Screen showing text data entry]

**Fig. 8.6**: Test sample collection using **Sampler**

The storage of sample data to a file overcame the problem of test subject availability to a large extent as they were only required to be available for testing on one occasion. Once test samples were obtained they could be run against any profile at any time (with the monitor still treating the stored data as if it was being entered in real-time), thus allowing greater flexibility in the testing process. As a result, there were very few missing test cases in the study.
8.4.2.3 Monitor

This compares user test samples stored by Sampler against the typing profiles generated by Profiler. The system plots an on-screen graph of the test subject performance against the profile, showing the percentage of invalid keystrokes and highlighting any points at which an authentication challenge would be issued. An example of this is shown in figure 8.7.

![Keystroke Authentication System (KAS) Test Subject Performance](image)

**Fig. 8.7** : Comparison of test sample and profile using Monitor

In terms of reporting, the Monitor details the peak values observed for consecutive invalid keystrokes and the percentage of invalid keystrokes, the total number of authentication
challenges issued and, if applicable, the number of keystrokes before the first challenge and the reason the challenge was triggered.

The system also generates and utilises a number of files, as listed in table 8.4.

<table>
<thead>
<tr>
<th>Files</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROFTEXT</td>
<td>TEXT</td>
<td>The text used by the Profiler module for the creation of user typing profiles.</td>
</tr>
<tr>
<td>TEXTANAL</td>
<td>REPORT</td>
<td>The character / digraph analysis of the current reference text.</td>
</tr>
<tr>
<td>PROFILES</td>
<td>DATA</td>
<td>The user keystroke profiles generated by Profiler and referenced by Monitor.</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>REPORT</td>
<td>The report entries created by the Profiler and Monitor modules.</td>
</tr>
</tbody>
</table>

Table 8.4: Keystroke analyser files

8.5 Full Keystroke Analysis study

Having identified the main issues considered in the design and implementation of the keystroke analyser, the discussion now proceeds to detail the experimental evaluation that was performed. This includes a description of the test subjects involved, the experimental procedure and the final results observed.

8.5.1 Test subjects

The analysis aimed to encompass subjects with a broad range of typing abilities. A total of 30 users eventually participated in the tests, ranging from professional typists to comparative novices.
It was not possible to get an equal distribution of subjects across the different typing skill categories as only a vague assessment of their skill level was possible prior to profiling (e.g. it was obvious whether someone was closer to being categorised “good” than “poor”, but impossible to distinguish between, say, “average (skilled)” and “average (non-skilled)” typists by simple observation). The actual skill level was assessed by the Profiler, by which time, of course, a profile had already been generated (and it was considered better to include all of these in the tests rather than allow any to be wasted).

A breakdown of the test subjects profiled typing performance and skill classification (according to the categories described earlier) is presented by the graphs in figures 8.8 and 8.9 and by the associated details in table 8.5 (which also lists the subject-specific thresholds that were established in each case).
Chapter 8: Real-time Supervision using Keystroke Analysis

Fig. 8.9: Test Subject Typing Skills

<table>
<thead>
<tr>
<th>Test Subject</th>
<th>Mean Key Interval (ms)</th>
<th>Standard Deviation (ms)</th>
<th>Typing Skill Classification</th>
<th>Subject-specific Challenge Thresholds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Max. % Invalid</td>
</tr>
<tr>
<td>1</td>
<td>109</td>
<td>51</td>
<td>Good</td>
<td>26</td>
</tr>
<tr>
<td>2</td>
<td>110</td>
<td>50</td>
<td>Good</td>
<td>32</td>
</tr>
<tr>
<td>3</td>
<td>121</td>
<td>47</td>
<td>Good</td>
<td>29</td>
</tr>
<tr>
<td>4</td>
<td>122</td>
<td>60</td>
<td>Good</td>
<td>30</td>
</tr>
<tr>
<td>5</td>
<td>148</td>
<td>67</td>
<td>Good</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>185</td>
<td>81</td>
<td>Average (skilled)</td>
<td>28</td>
</tr>
<tr>
<td>7</td>
<td>186</td>
<td>77</td>
<td>Average (skilled)</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>191</td>
<td>87</td>
<td>Average (skilled)</td>
<td>24</td>
</tr>
<tr>
<td>9</td>
<td>196</td>
<td>95</td>
<td>Average (skilled)</td>
<td>29</td>
</tr>
<tr>
<td>10</td>
<td>196</td>
<td>80</td>
<td>Average (skilled)</td>
<td>29</td>
</tr>
<tr>
<td>11</td>
<td>198</td>
<td>81</td>
<td>Average (skilled)</td>
<td>26</td>
</tr>
<tr>
<td>12</td>
<td>214</td>
<td>92</td>
<td>Average (skilled)</td>
<td>25</td>
</tr>
<tr>
<td>13</td>
<td>216</td>
<td>80</td>
<td>Average (skilled)</td>
<td>33</td>
</tr>
<tr>
<td>14</td>
<td>221</td>
<td>87</td>
<td>Average (skilled)</td>
<td>23</td>
</tr>
<tr>
<td>15</td>
<td>224</td>
<td>83</td>
<td>Average (skilled)</td>
<td>32</td>
</tr>
<tr>
<td>16</td>
<td>226</td>
<td>93</td>
<td>Average (skilled)</td>
<td>24</td>
</tr>
<tr>
<td>17</td>
<td>229</td>
<td>89</td>
<td>Average (skilled)</td>
<td>28</td>
</tr>
<tr>
<td>18</td>
<td>231</td>
<td>80</td>
<td>Average (skilled)</td>
<td>24</td>
</tr>
<tr>
<td>19</td>
<td>232</td>
<td>79</td>
<td>Average (skilled)</td>
<td>24</td>
</tr>
<tr>
<td>20</td>
<td>233</td>
<td>92</td>
<td>Average (skilled)</td>
<td>24</td>
</tr>
<tr>
<td>21</td>
<td>247</td>
<td>97</td>
<td>Average (non-skilled)</td>
<td>29</td>
</tr>
<tr>
<td>22</td>
<td>250</td>
<td>71</td>
<td>Average (non-skilled)</td>
<td>24</td>
</tr>
<tr>
<td>23</td>
<td>256</td>
<td>109</td>
<td>Average (non-skilled)</td>
<td>25</td>
</tr>
<tr>
<td>24</td>
<td>287</td>
<td>96</td>
<td>Average (non-skilled)</td>
<td>29</td>
</tr>
<tr>
<td>25</td>
<td>298</td>
<td>86</td>
<td>Average (non-skilled)</td>
<td>30</td>
</tr>
<tr>
<td>26</td>
<td>307</td>
<td>105</td>
<td>Average (non-skilled)</td>
<td>26</td>
</tr>
<tr>
<td>27</td>
<td>330</td>
<td>113</td>
<td>Average (non-skilled)</td>
<td>52</td>
</tr>
<tr>
<td>28</td>
<td>338</td>
<td>91</td>
<td>Average (non-skilled)</td>
<td>26</td>
</tr>
<tr>
<td>29</td>
<td>345</td>
<td>104</td>
<td>Average (non-skilled)</td>
<td>25</td>
</tr>
<tr>
<td>30</td>
<td>398</td>
<td>84</td>
<td>Poor</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 8.5: Profiled Performance of Test Subjects
Whilst the heavy concentration of “average skilled” typists may not be fully representative of a true user population, it does provide a good test of the system's ability to distinguish between typists of a seemingly similar nature. The lack of a “best” skill categorisation was surprising, especially given that trained typists were involved, and possibly indicates that Card's criteria is somewhat strict. It was considered legitimate that “poor” subjects were under-represented as these would be unlikely to found in the role of regular information system users anyway.

8.5.2 Experimental procedure

The experimental procedure adopted for the investigation was based upon the following stages.

1. A typing profile was created for each test subject on the basis of two samples of a 2202 character reference text (text 1).

2. Each subject entered two test samples, text 2 (574 characters) and text 3 (389 characters).

3. The text 2 and text 3 samples were used to determine the individual authentication thresholds for each legitimate user by running them against the profile. This allowed the peak ratings for the percentage of invalid keystrokes and number of consecutive invalid keystrokes to be obtained (see table 8.5).
4. Impostor tests were performed using both test samples to determine the FAR.

Each sample was compared against all profiles other than that of the legitimate subject who created it (giving a total of over 1700 impostor attempts).

Copies of the three texts that were utilised in the study (along with an analysis of their character and digraph composition) can be found in appendix D.

The use of the Sampler module to store text samples that could subsequently be replayed back through the Monitor on demand proved to be very useful, in that it allowed the experiment to be repeatedly remounted as new test subjects were added. This issue is one of several raised in a paper produced during the course of the research which discusses the applicability of computer simulation to the testing of security systems (Furnell et al. 1995a). A copy of the paper can be found in appendix F.

8.5.3 Results and analysis

With the FRR having been eliminated, the aims of the study were to determine the FAR and the speed of successful impostor detection.

In terms of overall impostor detection effectiveness, the experimental system exhibited a FAR of 15% across the two text samples, as shown in figure 8.10. However, given that each subject provided two test samples, it was also possible to investigate the level of impostor consistency. This was established by sub-dividing the test samples into the pairs that were generated by the same subjects and then determining the proportion of cases where both samples were able to pass as another user against those where only one attempt
was successful. This information is also illustrated in figure 8.10. It can be conjectured that, given longer test samples, the impostors who were successful in only one attempt would eventually be detected at some point (albeit after a more significant number of keystrokes) and that the overall FAR in practice might, therefore, be somewhat less. However, it appears unlikely that those who were successful in both cases would be detected within a reasonable time (if at all).

![Figure 8.10: Impostor detection performance](image)

In any case, the FAR observed represents only a slight degradation on the figure observed by Leggett et al (i.e. 13%), but without any associated false rejections.

The matrices in tables 8.6 and 8.7 present the full results of the study in respect to each of the sample texts. Columns represent test subject profiles and rows represent test samples (i.e. the impostor attempts). The subjects are again listed in order of typing ability, with the bold horizontal and vertical lines being used to denote the boundaries of each skill category.
Chapter 8: Real-time Supervision using Keystroke Analysis

For each conducted test, the grid squares indicate whether the impostor was able to successfully pass as the profiled user or, if not, why a challenge was issued (it should be noted that the majority of impostors were actually challenged many times during their test samples, but this is not reflected in the tables, which only indicate the cause of the first alert).

The key for both tables is as follows:

- **C** Challenge due to consecutive invalid keystrokes
- **%** Challenge due to percentage of invalid keystrokes
- **-** Test not conducted
- **Blank** False acceptance of an impostor

| Profile | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1       | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 2       | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 3       | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 4       | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 5       | C | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 6       | % | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 7       | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 8       | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 9       | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 10      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 11      | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 12      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 13      | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 14      | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| 15      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 16      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 17      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 18      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 19      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 20      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 21      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 22      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 23      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 24      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 25      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 26      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 27      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 28      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 29      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |
| 30      | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C |

Table 8.6: Overview of impostor detection for test sample 1
Chapter 8: Real-time Supervision using Keystroke Analysis

Table 8.7: Overview of impostor detection for test sample 2

It is immediately evident from these matrices that the FAR is greater in relation to the impostor attempts based on the second text sample. In total, 111 impostor attempts were successful using sample one, as opposed to 151 with sample two (i.e. a 36% increase in the second case). This can largely be explained by the fact that the text used was shorter than that for sample one and, therefore, less opportunity for impostor detection was presented. This substantiates the earlier observation that, with longer test sample texts, the FAR might be reduced.

Another observation resulting from the matrices is that the impostor detection performance of the two monitoring methods (i.e. percentage invalid and consecutive invalid keystrokes)
is very similar, with an almost exact 50% split between them in terms of the cases detected (actually 722 cases verses 724 !). As such, both methods can be considered to be useful authentication measures. As expected, no impostors were challenged as a result of there being insufficient data for analysis (however, this does not alter the requirement for such a safeguard to be included in practice).

From the matrices it was possible to extract the number of false acceptances in each row and column and thereby determine an overall summary of those scored by and against each subject. This is presented in table 8.8 and the accompanying chart in figure 8.11. For each individual subject, the percentages for false acceptance are based on 58 test cases (except in the case of subject 12, where only one test sample was taken and hence the percentage is based on 29 test cases), whilst the percentages for successful impersonation are based on 57 test cases (due to missing attempts by subject 12).

<table>
<thead>
<tr>
<th>Test Subject</th>
<th>Falsely Accepted</th>
<th>Successfully Impersonated</th>
<th>Test Subject</th>
<th>Falsely Accepted</th>
<th>Successfully Impersonated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td># cases</td>
<td>% cases</td>
<td># cases</td>
<td>% cases</td>
<td># cases</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>19</td>
<td>3</td>
<td>5</td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>26</td>
<td>1</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>33</td>
<td>1</td>
<td>2</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>17</td>
<td>1</td>
<td>2</td>
<td>19</td>
</tr>
<tr>
<td>5</td>
<td>16</td>
<td>28</td>
<td>1</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>19</td>
<td>33</td>
<td>8</td>
<td>14</td>
<td>21</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>14</td>
<td>22</td>
<td>39</td>
<td>22</td>
</tr>
<tr>
<td>8</td>
<td>11</td>
<td>19</td>
<td>27</td>
<td>47</td>
<td>23</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>12</td>
<td>21</td>
<td>37</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>19</td>
<td>6</td>
<td>11</td>
<td>25</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>17</td>
<td>2</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>10</td>
<td>15</td>
<td>26</td>
<td>27</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>21</td>
<td>23</td>
<td>40</td>
<td>28</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>21</td>
<td>23</td>
<td>40</td>
<td>28</td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td>14</td>
<td>12</td>
<td>21</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 8.8 : Figures for subject false acceptance and successful impersonation
A initial observation is that, in general, the subjects who performed best as impostors were the least easily impersonated and vice versa. Looking at the cases in more detail it can be seen that the more skilled typists were the most successful impostors and that the majority of false acceptances occurred where the impostor was of the same or better skill classification than the target profile.

It can also be observed that the test subjects who were most frequently and consistently impersonated (i.e. subjects 7, 8, 9, 13, 17 and 23) were generally those who had either larger than average valid keystroke ranges or high authentication threshold settings. Both of these factors would allow more leeway for impostors and, in either case, false acceptances could potentially be restricted by obtaining a more accurate keystroke profile for the affected users.
However, given that impostor detection was actually possible in the majority of cases, the next most important consideration was the speed with which it could be achieved (i.e. how many keystrokes was an impostor able to enter before being detected). The experimental findings on this aspect are shown in figure 8.12 below. This shows the percentage of impostors detected within five distinct keystroke ranges, with cumulative values also indicated.

![Graph showing keystrokes before impostor detection](image)

**Fig. 8.12 : Keystrokes before impostor detection**

These results indicate that the vast majority of impostors would be detected within 160 keystrokes (the equivalent of two standard lines of text), with detection in under 40 keystrokes in 26% of cases. Whilst this may not combat the most destructive scenarios (e.g. the immediate entry of “delete *.*” would very likely be unchallenged), it should be
sufficient to identify the more common types of intruder who generally require sustained access in order to effect a serious breach.

It should also be noted that these figures essentially characterise the impostor detection performance that would be observed from the point of initial login (i.e. beginning with 0% invalid keystrokes). However, in scenarios where an impostor takes the place of a legitimate user it is likely that detection would be quicker and more frequent, as a certain percentage of invalid keystrokes would already have been registered (by the legitimate user) and, therefore, the rejection threshold would be reached more easily.

The preliminary results from this system (based on a test group of 26 subjects) were published by Furnell at al (1995b) and a copy of the paper is included in appendix F. It should be noted that these results also yielded an overall FAR of 15%, showing some measure of consistency in the systems performance despite the subsequent addition of four further test subjects.

8.6 Potential Enhancements

A number of suggestions can be made for further development of the keystroke analyser. Several such ideas are presented below, but were considered outside the scope of this investigation (although significant other enhancement was performed, as will be discussed in chapter 9).
8.6.1 Impostor identity suggestion

A further (albeit possibly more ambitious) extension to the system would be to allow it to generate suggestions as to the most likely user(s) in the event of an impostor being detected / suspected (with appropriate confidence levels for each suggestion) - effectively altering the premise of the system from “are you who you say you are?” to “who are you?”. This could in theory be achieved by selecting the best match(es) from the reference profiles against the current subject. The information could then provide a basis from which subsequent security enquiries could begin. There is an obvious danger here that the system suggestions may provide misleading information, especially if the actual impostor was an outsider who did not have a reference profile stored by the system. However, the concept is more realistic when considered in the light of previous studies suggesting that the majority of security breaches are, in fact, perpetrated by insiders who are legitimately registered on the system.

8.6.2 Increased profile specificity

A number of ideas may be suggested for ways in which the keystroke-related aspects of IMS behaviour profiles could be made more specific:

- maintenance of different user / keystroke profiles to suit different applications or contexts (e.g. the profile for word-processing may be substantially different from that when using a database application);

- specifically analysing the keystrokes associated with commonly occurring words (e.g. "the", "and", "from" etc.) or key combinations that the user is known to
enter most frequently. This may overcome (to an extent) the fact that dynamic analysis is based upon arbitrary text input and that typing styles may generally be less consistent than with the regularly typed information that would be used in a static identifier. The premise is that more frequently occurring words would be more consistently typed and the approach would effectively apply a static identifier technique in a dynamic scenario;

- analysis of character “trigraphs” which might also be distinctive in some cases (e.g. “ING”, “THE”, “PRE”). These could be measured from the point of key depression for the first character until key release of the last.

8.6.3 Detection of subject impairment

Previous investigators (Joyce and Gupta 1990) have also suggested that an alternative application of keystroke analysis may be in detecting whether a legitimate subject is excessively tired or under the influence of alcohol / drugs (any of which would be likely to cause a noticeable departure from the "normal" profile). An obvious application for this would exist in safety critical environments such as healthcare, where it is generally important that subjects remain alert in order to deal with problems and minimise errors. Detection of the above is, therefore, advantageous based on the principle that a deterioration in physical performance may also be accompanied by a corresponding reduction in mental ability (a factor which could lead to errors of judgement and the like). It should, however, be noted that the end result of detecting an abnormality would be the same regardless of whether it was caused by an impostor or uncharacteristic behaviour from
a legitimate user (it would be extremely unlikely that a fine enough threshold could be established to enable differentiation between the two).

8.6.4 Neural Network implementation

The existing implementation is based entirely upon statistical methods, with the keystroke profiles being generated from a statistical analysis of the timings obtained during the profiling session. In subsequent monitoring, all profile entries are considered equal, with no distinction being made as to whether particular digraphs may be good or bad identity indicators (other than when the standard deviation exceeds the mean or insufficient samples were obtained).

It is believed that the system performance could be further enhanced by incorporating neural network techniques to analyse the user typing characteristics more closely. An inherent property of neural networks is that they have the ability to learn the differences between patterns, making them quite appropriate to the task of analysing the differences between typing styles.

Use of a neural network could enable the system to learn which typed digraphs are the most characteristic for each user (and, therefore, allow greater confidence to be associated with these when determining authentication judgements). This would also eliminate the need to set authentication challenge thresholds as the system would be effectively be determining them for itself.
The technique could also be used to enable the system to determine the optimal configuration for other aspects of monitoring, for example:

- determining how many recent keystrokes upon which to base the calculations for percentage invalid and percentage unusable keystrokes;

- determining the number of standard deviations from the profiled mean that a legitimate user can tolerate, possibly on a digraph-specific basis (so that the smallest possible valid inter-keystroke time range can be established for each user / digraph, thus further limiting impostor potential).

Several previous studies have successfully incorporated neural network techniques, although all have been in relation to a static verification approach (Brown and Rogers 1993; Obaidat and Macchiarolo 1992).

It should be noted that the keystroke analyser developed in this research programme is currently being modified as part of another project (Morrissey 1995). This will include the addition of neural network techniques and a paper discussing the findings (and contrasting them with this system) will be produced in the near future.

8.7 Potential Problems

This section considers a number of potential problems that may be encountered with keystroke analysis, relating to both conceptual and implementation issues. It should be
noted, however, that the significance of many points would be considerably reduced if keystroke analysis was just one of several mechanisms in a full IMS.

8.7.1 Consistency of users

The categorisation of typists into the six groups identified earlier tends to imply that the keystroke characteristics of specific individuals will always be consistent. This is not necessarily the case (e.g. under certain circumstances even a professional typist may revert to a "one finger" style - which could radically differ from their reference profile). In fact, one's ability to type and/or compatibility with the profile may be affected by various factors (some of which may be more easily compensated for than others):

- physical condition (e.g. injury to fingers)
  
  One of the most common concerns over keystroke authentication is something akin to "what happens if the subject's hand/fingers are injured such that the ability to type is impaired?". In such circumstances it may be desirable (or even necessary) to be able to bypass the continuous authentication system, as a radical departure from normal style could lead to constant requests to perform more explicit authentication. The ability to override the system in this way would be at the discretion of the security/system manager;

- illness or general fatigue;

- familiarity with current task/activity;

- interruption;

- concentration lapse;

- keyboard variations

The feel (quality) and layout of keyboards are often noticeably different between different systems/manufacturers. The results from the healthcare user profiling survey showed that
users frequently require access from a number of different locations / terminals. If the nature of the keyboards vary from one to another then it is likely that subtle performance differences may be detected by the monitor system at least until the user becomes more familiar with them.

- profile drift

In some cases user performance may depart from the profiled level as their typing skills improve or evolve over time. This will cause problems if profile refinement is not successfully incorporated in some way.

8.7.2 Mimicry

As with normal handwriting, it may potentially be possible for an impostor to forge the keystroke "signature" of a legitimate user.

A key issue if this were to be attempted would be the ability of the impostor to mimic the legitimate subject's typing characteristics with sufficient accuracy to fool the system. Common sense suggests that for mimicry to be feasible it would be necessary for the impostor to be possess the same or higher typing skill classification as the target.

The experimental results showed that poor typists were the easiest targets for compromise, generally being less consistent and exhibiting fewer characteristic rhythms in their keystrokes. Overall it is hoped that all but the total non-typist should exhibit some characteristics that are relatively unique. However, whilst observation shows that even the classic "one finger typist" may exhibit certain characteristics that may increase the difficulty of imitation (e.g. periodic "bursts" of speed over particular sequences of characters / words with which they are more familiar), such distinctive rhythms may be too infrequent (when compared to the rest of the typing) to contribute significantly enough to the authentication.
Successful mimicry in this manner assumes (at least to an extent) that the would-be impostor is in a position to know the typing ability of the target. If this was not the case, the impostor would be forced to experiment (increasing his / her likelihood of being detected). However, in many scenarios (e.g. amongst colleagues) the possibility of subject observation will exist and, therefore, informed mimicry cannot be ruled out.

8.7.3 User acceptance

As with other aspects of IMS, the issue of keystroke authentication (particularly in the case of continuous monitoring) raises the question of the acceptability to users. Will there be resentment of the idea (for example, on the grounds of it being too reminiscent of a "big brother" scenario) ?

The US National Institute of Standards and Technology (1993) has raised concerns over the legality of keystroke level monitoring in cases where the typed information is viewed (or recorded for later use by) system management. In this context it is rightly observed that monitoring would be analogous to an unauthorised telephone tap and the conclusion is that a system sign-on banner should specifically notify potential users that monitoring may take place (and that by using the system they are submitting to this policy).

It should be noted that the study described here would not raise these concerns as user keystrokes are not stored after analysis. In actual fact, the keystroke analysis concept is fundamentally different from some other forms of supervision in that it monitors how users are typing rather than what they type. Therefore, the sole purpose is clearly the protection
of the systems, resources and data, as opposed to a more wide ranging means of user surveillance. However, other potential aspects of IMS supervision (e.g. access time, application usage) would not necessarily share this distinction and, therefore, the sign-on banner is considered to be a sensible idea (if only in as much as it could further deter unauthorised use or system abuse).

A small survey of the test subjects involved in our practical examination revealed that the majority of them were comfortable with the idea of keystroke monitoring. The few who expressed doubts did so mainly on the basis that whilst the concept does not necessitate monitoring of work done and the like, it could potentially provide a basis for this purpose. An additional concern was the number of interruptions that it could cause with false rejections. A supplementary question in the survey asked how many false rejections each subject would be prepared to accept within the space of an hour. Answers to this varied, with subjects who only used computers occasionally being prepared to accept around three rejections, whereas more frequent users would only accept one.

8.7.4 Accuracy of keystroke timings

It is obvious that the accuracy of the timing system is an extremely critical factor in determining the success or failure of the concept.

It has been suggested by Joyce and Gupta (1990) and previous studies that problems may be introduced where the system is implemented in a time-sharing environment with access through a variety of networks, as this could render it impossible to obtain timings from the remote system of an adequate resolution.
This problem has actually been encountered in practice, during the monitoring of the "Wily Hacker" by Clifford Stoll at Lawrence Berkeley Laboratories (Stoll 1988; Hafner and Markoff 1991). At one point during his investigation Stoll, wishing to determine whether he was dealing with a single intruder or a group, decided that an analysis of the incoming typing rhythms could provide the answer. Stoll firstly set up an experiment within the confines of his lab, using his colleagues as the subjects, and discovered that it was indeed possible to differentiate between the different users. Suitably encouraged, Stoll proceeded to apply the test to the "keystrokes" coming in over the network. However, it soon became apparent that, as a result of the delays in transmission through intermediate computers, any information that could identify the typist had been lost (the data was simply received at evenly spaced intervals resulting from network transmission, with only occasional discrepancies being apparent when, for example, the hacker may have been searching for the next key).

It is, therefore, necessary for the timing data to be captured by the local keyboard and then subsequently transmitted for analysis upon request. In order to be viable, this would require either the use of intelligent terminals or additional devices to supplement dumb terminals to enable them to collect timings (the latter of which would probably negate any cost benefits that would be apparent from using the keystroke authentication concept). However, given that virtually all future terminals are likely to be provided in the form of PCs or workstations, obtaining the timings should not be a significant problem.
8.7.5 General applicability

Keystroke analysis cannot be regarded as a universal solution to the authentication issue. In some ways the concept may be deemed counter-intuitive, given the move towards "user-friendliness" of applications which in some areas can significantly reduce the role / requirement for the keyboard (e.g. use of keys only for simple menu-based selection, use of a mouse instead etc.). Therefore, in some applications, the periods in which the keys are used may be so brief as to make any measurements impractical / unreliable.

However, as mentioned in section 8.6.2, it may be possible to compensate in scenarios where the keyboard is still used to some extent by specifically profiling and monitoring users in relation to words or key combinations that are still known to be frequently typed. An example of this may be if a user has to issue a specific command or series of keys to invoke an application that is regularly used. However, it is acknowledged that such limited opportunities would significantly reduce the level of supervision possible with the technique.

8.8 Conclusions

It is obvious that, given the level of false acceptances observed, keystroke analysis cannot be regarded as a total panacea to IMS supervision requirements. Whilst it may help to combat abuse by penetrators, there are still clear areas which are not addressed (specifically, potential abuse by legitimate users and malicious processes).

That said, the practical study has served to illustrate the significant potential of the concept and indicates that keystroke data would still provide a perfectly viable profile characteristic in a large percentage of cases. The results observed are comparable with, if not better than,
the previously documented investigations. In addition, the pilot study was able to assess the potential discriminating power of typing characteristics other than simply inter-keystroke times, which had not been addressed in previous work.

Another advantage of the investigation was that it allowed the determination of a worst case FAR. However, it must, of course, be remembered that the consequent 0% FRR observed in the study was obtained artificially and some false rejections would be almost bound to occur in practice from time to time. However, with authentication thresholds set correctly, it is envisaged that these cases would not be frequent enough to significantly trouble legitimate users.

A FAR of 15% would be of less significance if the preliminary user identification phase was still to include some form of initial authentication (as suggested in section 6.5.2) as the combination would almost certainly serve to foil the majority of intrusion attempts. In addition, it should be remembered that this is only the FAR for one supervision technique - in a full IMS, using more comprehensive behaviour monitoring, the overall system FAR (based on a combination of approaches) could be significantly less.

As previously mentioned, it is considered that false acceptances could be reduced by generating more representative profiles of legitimate users. Whilst this would require larger text samples (which could be collected via a background process to reduce the user burden), it would potentially allow more accurate authentication thresholds to be set and reduce the number of unrepresented digraphs in the profiles (therefore allowing more keystrokes to be analysed).
In practice monitoring would also need to be implemented as a background process, analysing the keystroke data in real-time in conjunction with normal activities and remaining transparent unless an intrusion is suspected. The extension of the system to fulfil this role is the subject of the next chapter, which also goes some way to showing how keystroke analysis can be integrated into a more comprehensive demonstration of the overall IMS concept.
CHAPTER 9

An IMS Demonstrator System
9.1 Enhancement of the Keystroke Analyser

Whilst the results and analysis from the experimental study proved the effectiveness of the keystroke analysis concept, they did not demonstrate how the approach would be used in a practical context. Having established keystroke analysis as a suitable supervision technique, further development was undertaken to show how it could be incorporated into an operational security system, based upon some of the principles established in the IMS conceptual design from chapters 6 and 7.

The resulting IMS Demonstrator system is based upon the keystroke analysis "engine" from the experimental keystroke analyser, with extended functionality which also incorporates system configuration auditing utility, virus scanning and comprehensive system management options.

The system has been considerably extended to operate in a new configuration involving the use of two PCs communicating over a serial link, as illustrated in figure 9.1.
PC A acts as the monitored user workstation running the IMS Client. The Client is implemented as a transparent, background task collecting inter-keystroke timing data (see section 9.2.4). The user is initially identified by this machine and will be authenticated in the usual manner before the start of a supervision session. The system configuration of the machine is assessed at the start of each session (to ensure that it has not been compromised/tampered with in any way that might signify a breach of security), followed by a virus scan to detect the presence of malicious processes.

PC B is be used to run the IMS Host and holds the keystroke profiles for the registered users of the system, along with system configuration data relating to the Client workstation. The Host analyses all incoming inter-keystroke times from the local workstation, comparing them against the profile of the logged-on user (note that since no changes were made to the basic keystroke analysis approach, it was possible to re-utilise the user profiles from the experimental system). Discrepancies between the incoming times and the profile increase
the alert status for the current user session (which is maintained by the Host). This is sent as a continuous signal to the IMS Client.

Whilst the demonstrator does not set out to provide a full implementation of the IMS design, the minimum configuration is nevertheless sufficient to model the following:

- the Host - Client relationship;
- basic implementation of all IMS modules bar the Profile Refiner;
- transparent real-time user supervision (based on keystroke analysis);
- detection of external penetrator and masquerader-class intruders.

The system does not incorporate the use of generic intrusion rules or class-level behaviour profiles.

Aspects of the demonstrator will now be discussed in more detail, with descriptions of the Client and Host systems that have been implemented. It should be noted that these descriptions are not intended to act as either user manuals or technical reference - they provide an overview of the demonstrator functionality, highlighting aspects of the IMS design that have been addressed and significant additions to the basic keystroke analyser.
9.2 IMS Client Implementation

9.2.1 System Configuration Auditing

The activation of the Client can optionally trigger a system configuration audit of the local workstation, with the first task of the Host then being to verify the details collected (with immediate system administrator warnings and possible Client suspension if the configuration has been changed).

If system configuration auditing is enabled, details of the Client workstation's memory, disk and DOS set-up are transmitted to the Host for validation against stored details (in actual fact, the system audits all of the characteristics previously listed in table 6.2). Any discrepancies are then highlighted at the Host with a message describing what has changed (in addition, a more detailed description is written to a text report file, which also specifies the date and time of discrepancy detection and the original and changed configuration settings). The way in which the Client responds depends upon how the Host has been configured. If suspension is enabled, the Client session is locked to allow the Host system administrator to investigate the situation. The administrator must then answer two questions:

1. whether the configuration stored by the Host should be updated;
2. whether the Client session should be unlocked.
Chapter 9: An IMS Demonstrator System

The Host then responds accordingly (note that if the Client remains locked, the only option is to reset it and, unless the configuration details stored by the Host have been updated, suspension will keep occurring until the original system configuration is restored).

File checksum calculation is limited to the CONFIG.SYS and AUTOEXEC.BAT files if they exist on the Client system. These are considered the most important files as far as system integrity is concerned and their inclusion serves to demonstrate the principle effectively enough. The checksums themselves are based upon a simple function of the size and composition of the target files.

In a multi-workstation implementation IMS would need to store several configuration records and each workstation would consequently have its own unique identity that would be transmitted to the Host to indicate which record to use. This element has not been implemented in the demonstrator given that it is limited to operating with a single client workstation anyway.

9.2.2 Virus Scanning

Client activation may also optionally invoke a virus scan of the local machine, depending upon the Client Control and Virus Scanning options selected at the Host.

If scanning is enabled, the Client executes an external DOS program to handle the operation, whilst the Host waits for the completion status of the operation to be sent. If viruses are detected then, as with configuration audit discrepancies, the Host can optionally suspend the Client to allow the anomaly to be further investigated.
The scanning is based upon the shareware virus checker F-PROT (Skulason 1993), which is called by the Client process using parameters passed to it by the Host. These parameters correspond to command line options offered by F-PROT.

F-PROT is an example of one of the most widely used approaches to virus detection and works by scanning memory and nominated groups of files for specific byte patterns extracted from each known virus (this pattern is referred to as the *virus signature*). It has the ability to recognise 818 families of virus, with each family consisting of anywhere between 1 and 150 viruses (giving a claimed total of up to 2933 viruses). A number of scanning options are offered, the following of which are supported by the demonstrator (options shown in bold are mandatory, whereas the remainder may be selectively enabled by the Host administrator):

```
/ALL      Check all files.
/DELETE   Delete all infected files.
/DISINF   Disinfect whenever possible.
/HARD     Scan all DOS partitions on the hard disk.
/NOBREAK  Do not abort scan if ESC is pressed.
/NOMEM    Skip initial memory scan.
/[NO]BOOT [Do not] scan boot sectors.
/[NO]FILE [Do not] scan files.
/[NO]PACKED [Do not] scan inside packed files.
/[NO]TROJAN [Do not] scan for trojans and joke programs.
/[NO]USER  [Do not] scan for user-defined patterns
/Old      Do not complain if the program is outdated.
/REPORT=  Send the output to a file.
/SILENT   Don't generate any screen output.
```

Of the further parameters offered, the following are not supported as they are not applicable to the demonstrator context and / or would serve no useful purpose:

```
/COMMAND Force command line mode.
/EXT=    Specify default file extensions to scan.
/HELP    Display this list.
/MONO    Use monochrome mode on color displays.
```
Finally, the inclusion of the remaining options was considered undesirable as their use could potentially reduce scanning effectiveness or produce unwanted results:

- **/640**: Only scan 640K of memory.
- **/ANALYSE**: Use heuristic analysis, instead of signatures.
- **/APPEND**: Used with /REPORT - append to existing report.
- **/AUTO**: Automatic deletion/disinfection.
- **/LIST**: List all files checked.
- **/QUICK**: Faster search, but not as accurate.
- **/RENAME**: Rename infected files to *.VOM or *.VXE.
- **/NOSUB**: Do not scan subdirectories.

It should be noted that as the demonstrator uses F-PROT version 2.09 (circa July 1993), it is possible that (a) certain virus strains will not be identified and (b) detection options available in later versions of the software are not supported. As such, it may be necessary to modify the system if a more recent version is used in order to take advantage of any new or changed command line options.

A further consideration arises from the fact that F-PROT is an external DOS program and, hence, returns no status value upon termination. This prevents it from being able to directly report the results of the scan back to the Client (which, in turn, constrains the Clients ability to report to the Host). As a result, the Client must determine for itself whether F-PROT detected any anomalies. This necessitates that a rather untidy solution be used, whereby the Client parses a report file created during the scan (note: this file is subsequently deleted...
unless report generation was specified by the Host administrator). It was established that if no viruses are detected, an F-PROT message to this effect always appears as the last entry of the report file (a fixed number of bytes from the end-of-file). The Client process, therefore, reads the file from this position and looks for the start of the message. If it is not found then it is assumed that one or more viruses were detected and the Host is notified accordingly.

Whilst this overcomes the basic problem of reporting a virus scan result back to the Host, a weakness is that the capability still only extends to stating whether or not any anomalies were found. For specific identification of the number and type(s) of viruses involved the Host administrator must still examine the F-PROT report file at the Client.

### 9.2.3 User Identification and initial Authentication

Before supervision can commence, the demonstrator must perform a user identification phase so that an identity can be sent to the Host. This is required to enable the Host to determine which profile should be used and, as such, this stage is mandatory and cannot be disabled.

In addition to requesting an identity, the opportunity has also been taken to perform basic authentication (via a simple password) as an additional safeguard. It should be evident that the combination of this stage along with the system configuration audit provides a means of verifying both the integrity of the local workstation as well as the legitimacy of its user prior to the start of the session proper.
Despite the discussion in section 6.5.2, the process is based upon a standard login procedure rather than the more advanced faceprint or voice verification techniques. These facilities were not available within the development environment and, in any case, the demonstration of this aspect was regarded as secondary to the need to show real-time supervision in operation.

A disadvantage of the software-based PC implementation is that the Client is currently executed from the AUTOEXEC.BAT file of the local system. As a consequence it is possible for the user to prevent the Client from being installed (and thereby bypass supervision) by booting from the floppy drive or interrupting the AUTOEXEC sequence. In a full implementation this would have to be prevented and a potential solution is outlined in section 9.4.2.

9.2.4 Implementation and operation of background supervision

The transparent background operation of the Client is achieved using a special form of MS-DOS coding, called a Terminate and Stay Resident (TSR) program (Angermeyer et al. 1989; Christopher et al. 1990). The difference between these and normal applications is that some of the code is retained in memory after program termination (and can then be activated later by a variety of stimuli). The programs can, therefore, be divided into two portions: the initialisation code and the resident code.

The purpose of the initialisation code is to perform any initial functions that may be required and set up the environment for TSR operation. In terms of the IMS Client, this involves the following key stages:
1. check for previous TSR installation;
2. perform all other foreground processing (i.e. set up serial communications, perform system audit, virus scan and user identification operations);
3. install replacement interrupts (i.e. key action, timer tick and equipment list);
4. terminate and stay resident (retaining functions required for issue of challenges, session suspension etc. as the resident code).

Stage one in the above is achieved using the largely redundant DOS equipment list interrupt (0X11), which is patched the first time the Client code is run. Whenever it is run, the first thing that the Client does is to call the equipment list service, passing a value 1111 in the CX register. If the TSR is already installed, then the aforementioned patch will cause 2222 to be returned in CX as a response. If this value is not returned, the Client process knows that it is okay to proceed with TSR installation (note that any calls to equipment list by other applications are unaffected as the patch preserves the normal data which it returns).

Subsequent activation of the TSR is then triggered by the key action and timer tick interrupts, so it effectively operates continuously throughout the remainder of the user session. The amount of processing performed by each of the interrupts is negligible, so applications still operate as normal and the fact that the TSR is installed will remain transparent unless an impostor is suspected.

During background operation the Client monitors signals received from the Host to determine whether it must perform any action to respond to a suspected intrusion. These
signals equate to a basic version of the alert status from the full design. The signals currently used are:

- all clear;
- issue challenge;
- lock workstation;
- unlock workstation.

The latest alert status signal is read by the Client every time a key is pressed, effectively allowing an immediate response to any anomalies.

A continuous signalling method is used such that if no specific Client action is required, the Host sends out an “all clear” signal. This is used as a means for the Client to determine the integrity of Host operation - i.e. if no signal at all is received then the Client assumes that the Host has been compromised in some way and can automatically lock the local workstation. The session is then subsequently unlocked if / when the Host becomes operational again and the status signal resumes. However, it is noted that in some scenarios it may be preferable to accept the risks of an unsupervised session rather than allow the workstation to be locked (as the enforced unavailability of a vital system could have as equally serious consequences as an impostor breach) and, as such, the system can be configured not to do so (however, the Host must always be operational at the start of the Client session in order to perform the configuration audit and user authentication).
A regular signalling system is also implemented in the other direction, in that, at all times when keystroke data is not being transmitted, the Client sends a regular (once per second) signal to the Host to indicate that it is still operational (i.e. the Client machine has not crashed or been reset). This is used to allow the Host to detect user "logout", as described in section 9.2.6.

9.2.5 Challenges and Session Supervision

As previously mentioned, the demonstrator maintains a basic alert status which is sent to the Client TSR. Two levels of action may potentially result as the alert status increases:

1. the Host automatically initiates the issue of a challenge to the user at the monitored workstation. The challenge is based around a question and answer password (using personal information supplied by the user during profiling);

2. the Host causes the Client TSR to suspend the session at the monitored workstation.

Additionally, the Host administrator is able to manually cause a challenge to be issued or suspend the session if he / she is suspicious of the Client user for some reason.

The issue of a challenge by the Host is registered at the Client after the first subsequent keystroke, with the challenge window being displayed on the second (allowing virtually no opportunity for impostor action after anomaly detection). This sequence is illustrated in
figure 9.2 below, showing an example impostor input and the IMS actions that would occur after each keystroke was made.

Fig. 9.2 : Response to challenge signal

9.2.6 User logout / Session termination

In addition to identifying the user at the start of a session, a further requirement is for the Host to be able to detect session termination (for auditing purposes and so that it can be reset to allow a new monitoring session to commence).

The initial plan was to allow users to explicitly logout from the Client (e.g. via a hot-key combination), which would then notify the Host and reset the TSR to allow the next user to login. However, a flaw here was that the issue of a machine reset (e.g. ctrl-alt-del) on the Client PC would result in the removal of the TSR without Host notification and monitoring would have to be reset manually. Attempts to overcome this proved unsuccessful as no
readily documented means was found to detect (and then subsequently avert) a ctrl-alt-del. It was, therefore, necessary to adopt a different approach.

The solution employed is that the Host keeps a constant watch to ensure that the Client is still operational, as opposed to the Client having to tell the Host that it has been reset,. This was achieved by having the Client send out regular “dummy” signals (once per second) if no keystroke data is being transmitted. These signals are filtered out and subsequently ignored by the Host, but serve the purpose of proving that the Client TSR is still running.

As a result, the Host should, under normal circumstances, always be receiving some kind of signal (keystroke data or “dummy”) every second and if this is not the case it can be assumed that the Client system has been reset. However, in the implemented demonstrator the Host actually waits for 10 seconds of inactivity before assuming that a logout has occurred. The reason for this is that the launch of some applications (e.g. DOS EDIT) will cause the Client clock rate to be reset to the normal 18.2 ticks per second (as opposed to the 1000 ticks rate used for inter-keystroke timing). This in turn delays the transmission of the “dummy” signals, meaning that the Host will temporarily not be receiving one every second. The 10 second period allows enough time for the Client TSR to have rectified the situation by reselecting a high tick rate and, therefore, resuming regular signal transmission.

The use of this approach meant that it was not necessary to provide specific “logout” functionality in the Client - users can simply reset or switch off the machine when they have finished.
Chapter 9: An IMS Demonstrator System

9.3 IMS Host implementation

9.3.1 Menu Options & General Functionality

The Host is predominantly menu-driven and provides a comprehensive range of security control options, as shown in figure 9.3 and described in more detail below.

Fig. 9.3: Demonstrator menu structure

9.3.1.1 Monitoring Options

Monitor Workstation

- force a remote challenge (e.g. used if security manager is suspicious of the current user);
- force a remote lock (for the same reasons as above);
- unlock remote system (i.e. after session suspension);
- configure real-time graph display;
- monitoring controls (on, off and reset).
Monitor Setup

- Configuration options affecting monitor sensitivity (i.e. highpass filter level, valid standard deviation range, number of recent keystrokes to monitor).

- Configuration options affecting monitor operation (i.e. whether continuous monitoring is enabled and, if not, how many valid keystrokes must be entered before it is suspended and what period of user inactivity will cause it to be resumed).

9.3.1.2 Profiling Options

User management

- list registered users;

- user registration (i.e. keystroke profiling);

- test profile (establishes user-specific authentication thresholds through collection and analysis of two text samples);

- profile update;

- profile deletion.

Text management

- text creation (including automatic text analysis);

- text editing (based on the DOS EDIT utility);

- text analysis (for independently created or modified files).

Profiler Setup
Configuration options for number of sample iterations and profiler highpass filter level.

9.3.1.3 System Management

Keystroke monitoring control;

Auditing

- inspect audit log;
- configure auditing;
- clear log entries.

Client control options (i.e. to determine how the Client behaves).

- system configuration auditing enabled (y/n);
- suspend Client if configuration modified (y/n);
- suspend Client if Host signal lost (y/n);
- Client virus scanning enabled (y/n);
- suspend Client if virus(es) detected (y/n).

Virus scanning options (i.e. F-PROT parameters enabled).

Intrusion action control (e.g. to determine how the system responds to suspected intrusions).

- maximum warnings before challenge issue;
- maximum failed challenges before suspend;
Chapter 9: An IMS Demonstrator System

- monitoring reset / time-out period.

View Client Setup
Displays the Client system configuration details currently stored by the Host.

9.3.2 Intrusion Monitoring and Detection
As with the experimental system, the Demonstrator identifies suspected intrusions by monitoring three factors:

- % invalid keystrokes;
- consecutive invalid keystrokes;
- % unusable keystrokes (i.e. inter-keystroke times exceeding highpass filter).

The Keystroke Monitoring options allow the Host administrator to specify which combination of these methods should be in operation during the monitoring session. In theory, all options should be enabled at all times for maximum impostor detection potential. However, in practice some methods may be found to cause problems for legitimate users (and would, therefore, be better disabled until a more adequate typing profile could be obtained).

In common with the description of the IMS Anomaly Detector in section 7.4.1, the demonstrator includes an option for supervision to be automatically suspended and resumed during user sessions (rather than operating continuously at all times). The number of keystrokes required before authentication is granted and the subsequent period of user
inactivity after which monitoring will resume are both configurable options. The system default in the first case is 300 keystrokes, which is considered to provide an adequate window for supervision - especially given that the experimental study showed that 160 keystrokes was normally sufficient to trap the majority of impostors. The default for the resumption of monitoring is after three minutes of inactivity. Options also exist for the Host administrator to manually control monitor operation (for use in cases where he / she may be suspicious of a user or, alternatively, confident of their legitimacy).

The incorporation of this idea (which was not part of the experimental system) is seen as having two main advantages in the practical context:

- it helps to further minimise the likelihood of false rejections;

- it will (to a limited extent) allow for the fact that users may sometimes entirely depart from their profiled typing style (for example, by adopting a one finger approach). Provided that authentication is already been determined by the initial keystrokes, users will be permitted more leeway in how they subsequently behave.

In order to avoid confusion, the Host reports whether continuous monitoring is enabled or disabled as the first task once supervision begins.

During monitoring sessions, a real-time graph is displayed showing the percentage of invalid keystrokes against time. If the percentage of invalid keystrokes are being monitored, the
current users authentication threshold is shown on the graph as a horizontal red bar at the appropriate level. Under normal circumstances, points on the graph are plotted in white. However, there are two exception cases, as follows: (a) if consecutive invalid keystrokes are being monitored, graph points denoting invalid keystrokes are plotted in green and (b) if monitoring is suspended then points are plotted in black (just to indicate when keystrokes are received). The issue of a challenge is denoted by a vertical bar, colour-coded to indicate the challenge cause as follows: red (% invalid keystrokes), green (consecutive invalid keystrokes), blue (% unusable keystrokes) and yellow (issued manually).

In addition to the graph, a status report window is provided which displays date / time-stamped messages relating to security-relevant events. The events reported include anomaly detection, issue of challenges, challenge failures and suspension or unlocking of Client session. An example of the principal monitoring interface is shown in figure 9.4.

Fig. 9.4 : IMS Demonstrator, Keystroke Monitoring Interface
9.3.3 Event Auditing

The Host features comprehensive event-configurable auditing to allow a log of all "security-relevant" incidents to be maintained. The events that can be recorded include:

- login failures;
- start and end of user sessions;
- anomaly warnings;
- issue of impostor challenges;
- challenge passes and failures;
- suspension and unlocking of client sessions;
- registration of new users (profiling);
- update of user profiles;
- user profile deletion;
- change of profiler "reference text";
- start-up and shutdown of IMS Host;
- changes to the Client workstation system configuration;
- results of Client virus scans;
- suspension and resumption of monitoring during sessions.

In each case the date and time of the event is recorded, as well as the identity of the user involved (if applicable). As indicated above, limited features for managing the collected data are also available.
9.3.4 Internal Communications

An overview of the Hosts internal communication flows and file usage (i.e. excluding any interactions with the Client) is shown in figure 9.5.

The configuration file shown in the figure is used to store various settings and user preferences relating to system operation. These currently include:

- profiler settings (i.e. number of reference text sample iterations, highpass filter level);
• monitor sensitivity settings (i.e. highpass filter level, valid standard deviation range, number of recent keystrokes used to calculate monitor statistics);
• monitor operation settings (i.e. continuous monitoring enabled / disabled, thresholds for suspension and resumption of monitoring);
• intrusion detection preferences;
• auditing preferences;
• client control preferences;
• virus scanning preferences;
• intrusion action thresholds;
• real-time graph display preference.

9.3.5 Host - Client Communication

An overview of the data exchanges between the IMS Client, running on the local workstation, and the IMS Host system is shown in figure 9.6.
Chapter 9: An IMS Demonstrator System

Note that the user passwords are transmitted to the IMS Host for validation rather than being checked locally within the Client. This ensures that the valid responses always remain in the secure Host (and thus reduces opportunities to compromise security by monitoring the communications link). However, the transmission of valid responses from the Client could still be vulnerable (unless encryption is employed).

9.3.6 Profiling Sub-System

The demonstrator provides facilities for handling user profiling. This includes both user registration and text management features, as outlined in figure 9.7.

The profiler reference text (from which the initial user profile is created) and two test sample texts (from which the user-
specific authentication thresholds are established). A feature is provided to allow new texts to be entered from within the system or, alternatively, text may be used from an existing file created elsewhere. The system generates a statistical analysis of all new texts, detailing the frequency of occurrence of individual characters and the 70 most frequent character digraphs. In addition, a series of text suitability ratings are calculated in order to provide a measure of a text's usefulness as the basis for profiling. The ratings are based upon the degree to which the 30 most common English language character digraphs are represented within the text. Four ratings are generated in total, as listed below:

- Rating 1: represents the percentage of the 10 most common digraphs that occur 5 or more times in the text;
- Rating 2: represents the percentage of the 11th to 20th most common digraphs that occur 5 or more times in the text;
- Rating 3: represents the percentage of the 21st to 30th most common digraphs that occur 5 or more times in the text;
- Rating 4: represents the overall percentage of the 30 most common digraphs that occur 5 or more times in the text;

Note that the digraphs are required to appear at least 5 times in order to ensure that a reasonable number of samples would be taken during the formation of any resulting profiles. The higher the ratings then the more appropriate a text should theoretically be as a basis for user profiling.
The separate Text Analyser option is provided to generate statistical analyses and suitability ratings for text files that have been created independently of IMS (or for files that have been modified - see below).

Whilst not shown in figure 9.7, a text editing option is also provided to allow modifications to existing reference and sample texts. This makes use of the MS-DOS EDIT utility and is basically intended as a means for mistakes to be corrected or for other small changes to be made (the reason being that, due to a restriction in the demonstrators text handling, no linefeeds can be included in any of the texts it uses and EDIT, therefore, displays the whole text as one continuous line. Note that if any linefeeds are included they are filtered out by IMS before the text is used).

9.4 Implementation constraints and potential enhancements

Setting aside the aspects of the IMS design that are not addressed, it is important to recognise that the demonstrator system still suffers from a number of limitations that would potentially reduce its usefulness in a practical context.

9.4.1 Limited intrusion detection

Possibly the most significant constraint of the demonstrator is that real-time intrusion detection functionality has been limited to the keystroke analysis technique discussed in chapter 8. The reason for this, as identified at the start of that chapter, was the relative ease with which keystroke profiles could be developed in comparison with other supervision techniques. The fact that the demonstrator was developed by an individual project rather
than a team effort meant that there was insufficient scope for practical examination of other
techniques (although scope obviously exists for future research efforts in these areas).

9.4.2 Insecure Client start-up

As previously mentioned, the nature of the PC / DOS environment currently handicaps the
Client initiation process in that it is possible to bypass the supervision system by either
booting from the floppy drive or interrupting the hard disk boot-up.

This weakness could be overcome by the use of secure hardware to force the system to
boot into the IMS Client before allowing any user interaction. The principal stages of the
PC start-up procedure can be described as follows:

1. automatic Power On Self Test (POST);
2. system disk boot-up (with priority given to floppy drive);
3. execution of CONFIG.SYS file - uninterruptable;
4. execution of AUTOEXEC.BAT file - interruptable.

If the floppy drive was temporarily disabled following the POST and the Client system then
initiated at the CONFIG.SYS stage rather than in the AUTOEXEC file, it would be
impossible for the installation of the supervision system to be circumvented.

It would, in fact, be possible to effect this solution by installing a ROM adapter card in the
Client PC. This ROM could be positioned in memory in such a way that it would be
executed immediately after the firmware for the POST, before any floppy drives are
recognised (Shepherd 1992). When execution is passed to it, the ROM would disable the floppy drive via software methods, forcing the system to boot from the hard drive. The Client software would then be installed as a *device* in the CONFIG.SYS file, and would include appropriate code to re-enable the floppy drive once supervision was in operation. In this manner, the Client could not be bypassed without the need to physically open and modify the computer.

9.4.3 Single workstation monitoring

Another fundamental departure from the full IMS design is that the demonstrator is only intended to monitor a single workstation / user session at any one time (although there may, of course, still be a significant number of registered users).

It is considered that the concept would be most usefully implemented in the context of a Local Area Network (LAN) environment, where the single IMS Host would be responsible for monitoring users on a network with many Client workstations. A basic idea of how this could be approached is described below.

Rather than continually transmitting activity data, the local Clients would collect and accumulate information (storing it in a temporary file) and then transmit it to the Host (e.g. via FTP using a specially created account and automated login) as a block representing X minutes of Client activity. The Host will then analyse the data received and immediately send an alert to any Clients with anomalous activity. This is illustrated in figure 9.8 (note that a Star topology is shown for ease of illustration only).
Fig. 9.8: Polling of active IMS Clients in network monitoring system

The obvious drawback here is that the impostor detection will no longer be occurring strictly in real-time, but will be delayed until the relevant Client is polled and its data analysed by the Host. As a result, the polling frequency will be crucial in preventing too great a window of opportunity for impostors. This will, however, be constrained by the following factors:

- the network size and the number of active Clients in operation;
- the typical network traffic loading (frequent polling could overwhelm network with IMS-related traffic);
- the speed at which the Host can analyse and, if necessary, respond to the incoming data from each Client.
Chapter 9: An IMS Demonstrator System

It would be desirable for all Clients to be closely monitored at the start of a session to ensure that an impostor is not active from the outset. However, after this the desirable polling frequency depends upon the type of intruder that one wishes to detect. For example, polling a Client once every 5 minutes may be enough to effectively detect penetrators (given that some delay would be likely between a legitimate user leaving and an impostor becoming active), whereas such an interval would be inappropriate for misfeasors as anomalous activity could start at any time.

In a full implementation of IMS, the potential problems here could be limited by transferring some of the more straightforward intrusion detection functionality (e.g. identification and response to login failures, suspicious command sequences) from the Host to the Client systems. This would give the Clients a degree of independence and leave them less vulnerable between polling cycles. The functionality transferred would most likely to relate to the generic intrusion rules as opposed to the monitoring of user behavioural characteristics. In this scenario, data would still be accumulated and analysed by the Host as originally described but, in addition, Clients could also specifically request Host attention if any anomalies were detected in the interim periods.

As an example of the quantity of data that would be involved in each Client-Host transfer, the Keystroke Analyser currently sends three bytes for every keystroke entered at the local workstation (two for the inter-keystroke time + one for the typed character). A fast typist may manage around 400 keystrokes per minute (giving a total of approximately 1.2K per minute in keystroke information alone). Given that keystroke analysis would probably be
the most significant element in terms of its data transmission requirement, and that details of some other activities (e.g. OS commands, application usage) would be encapsulated in that data as well, the largest Client-Host packet size would be probably be 2-3K per minute.

In this scenario it will also be important to safeguard the integrity of the temporary files created by the local Clients to prevent them from being modified (or deleted) prior to transmission to the Host.

9.4.4 MS-DOS implementation

The MS-DOS platform was retained for the demonstrator due to the existing experimental implementation and the easy availability of DOS-based software development tools.

However, most PCs no longer run DOS-based application software and have moved on to the more user-friendly platform offered by Microsoft Windows. Unfortunately, the demonstrator in incompatible with this environment, with virtually all aspects of implementation requiring modification to rectify the situation:

- obtaining keystroke timings;
- co-operation with other applications;
- serial communications;
- user interface design.

However, the DOS implementation does not prevent intrusion detection from being demonstrated in a practical context (e.g. using a DOS application such as a wordprocessor)
9.4.5 Code design and development

Although the demonstrator has been coded in a structured manner and incorporates functional elements from virtually all of the IMS modules, the development of the system did not adhere strictly to the modular structure proposed in the IMS conceptual design. The principal reason for this was that the foundation of the demonstrator was provided by the code from experimental keystroke analyser and the subsequent extension was geared around this framework. As such, enhancement of the system to incorporate further supervision techniques would be more difficult, involving modification to aspects of several code modules, relating to both Host and Client functionality.

9.5 Conclusions

Despite the limitations identified, the demonstrator does achieve its objective of proving the workability of real-time monitoring in practice and illustrating how it can be incorporated into the context of an operational security system. The system in its current form is also believed to represent an advancement on any previously documented investigation of dynamic keystroke analysis.

On a practical level, further development was constrained by the time available within the research programme and the need to address other aspects of the work in addition to the demonstrator. In addition, suitable Windows development tools and LAN facilities were unavailable within the research environment, effectively preventing these aspects from being pursued. However, even if such facilities had been available, addressing the issues of Windows and LAN implementation would not have represented research so much as
straightforward software development, which would ultimately have been unlikely to affect
the end results / findings relating to the effectiveness of the system.
CHAPTER 10

A Wider Framework for Healthcare Security
10.1 Inter-domain communications in healthcare

Having covered the practical elements of the study, the discussion now proceeds to examine healthcare security needs on a wider scale. The IMS concept (in conjunction with the CISS architecture discussed in section 7.7.1) is considered to be an appropriate means of providing comprehensive security within individual domains, allowing complete mediation of user activity. However, whilst it is likely that the majority of secure operations will still be restricted to the local domain, it is also necessary to consider the security of inter-domain operations, given the increasing requirement for transfer and exchange of data between HCEs (on a potentially international scale). In fact, a number of future trends involving inter-HCE communications have been predicted (European Commission 1994):

- increased inter-HCE networking;
- increased exchange of data between HCEs;
- increased potential for sharing of facilities between HCEs;
- establishment and adoption of the composite electronic healthcare record (EHCR).

Steps to realise these objectives are already in progress. For example, the UK National Health Service has already planned to bring all aspects of voice and data communications together into a common framework, with all major HCEs having the facility to communicate electronically by 1996 (NHS 1992). In addition, European project sponsorship is underway at the time of writing that will encourage and speed this progression. In these scenarios the key issue is likely to be that of trust between the
participating establishments - a factor that cannot always be guaranteed between communicating parties, even in healthcare. This places renewed emphasis on the need for security, with key issues being integrity (of both services and data), non-repudiation of activities and confidentiality. This is necessary not only to prevent unauthorised or undesirable activities, but also to provide a level of trust that allows broader and better services to be introduced.

In general terms, the demand for secure inter-domain communications in healthcare can be closely linked to three main factors:

- the increasing mobility of patients within the European healthcare community, with a consequent need to exchange healthcare records;

- the increased networking, accessing and sharing of systems between HCEs;

- the increasing desirability and viability of telemedicine services (i.e. medical diagnosis and treatment conducted at a distance from the patient).

The requirement to share healthcare records is largely a result of the increasing integration within the European Union and the choice that is consequently offered to patients. However, the viability of the idea is dependant upon the records being both portable and accessible, which in turn dictates a requirement for security. The portability issue is overcome by the emergence of the electronic healthcare record (EHCR), but the new opportunities that this offers also introduce some additional security concerns even before
the issue of inter-domain exchange is considered. For example, the nature of the information held in records is changing to include the integration of images, voice and data in a multimedia framework (Arnold and Peter 1993). This introduces a further concern in that the amalgamation of different forms of data into the composite record may potentially increase the sensitivity of the information beyond that of any of the component parts. Transmission of the records over a network only serves to heighten the concern and it is recognised that up until now much of the protection of HCRs has been provided by the fact that they rarely left the originating establishment and could not be accessed from external facilities (Barber and O'Moore 1991).

Sharing of HCE systems is desirable in that it could enable establishments with limited resources to overcome their lack of facilities. However, additional consequences of this will be increased complexity and interdependence of healthcare information systems and an increased access control burden. Significantly more people will have the potential to access (parts of) the system and they will no longer be confined to members of a single establishment.

As inter-HCE communication becomes more technically feasible, it is likely that the desirability and the potential applications of the service will increase. Widespread networking will enable easier and more effective communication between hospitals, laboratories and community care establishments and increase the level of potential contact between specialists and generalists. In short, information and expertise that may be lacking in one establishment should become more easily available from other HCEs. The combination of this point with the earlier one regarding the sharing of facilities illustrates a
way in which inter-HCE communications could be used to somewhat offset the potentially
damaging trend observed by Barber (1991a) in chapter 2, where healthcare knowledge and
resources become unequally distributed within Europe, leading to consequent inequalities in
the level and quality of services available in some establishments.

Finally, such networks would allow an opportunity to broaden the possibilities in medical
care. Some specific examples could include the following (Pfitzmann and Pfitzmann 1991):

- monitoring of patients in their homes;
- real-time video transmission during operations;
- accessing of large databases (e.g. cancer registers) for research purposes;
- provision of medical advice databases for consultation by patients.

The sensitive nature of healthcare and the systems involved will demand that all such
activities and exchanges can occur securely, with the properties of confidentiality, integrity
and non-repudiation all being of potential importance. Again, whilst these concerns are
recognised, it does not necessarily mean that they have been properly addressed. For
example, the aforementioned NHS-wide network has already been criticised by the British
Medical Association for having lax security arrangements (The Times 1995).

A further observation is that, at the national and international levels, healthcare
communications will utilise shared networks alongside data from other fields (e.g. over the
Internet). The communications infrastructure as a whole will, therefore, be a rich target in
terms of various types of potentially sensitive information; which serves to increase the risks
associated with any of the individual types of data being communicated. For example, healthcare data may be left vulnerable as a result of attacks targeting other types of information on the network (e.g. banking or governmental communications). Healthcare establishments must therefore have a means by which the security of their exchanges can be maintained in this scenario. However, a suitable protection strategy can be specified as the next logical extension of the IMS and CISS architectures that have already been discussed.

10.2 Enabling secure inter-domain operations

Suitable methods for achieving the required services on a wide scale are largely based around cryptography and involve the use of digital signatures, data encryption and the support of Trusted Third Party (TTP) infrastructures. These will be described in the sections that follow. However, an exhaustive technical analysis of the techniques is not provided as suitable references are subsequently included in the text. The intention is to illustrate how the concepts may be used to extend the security framework previously described to enable secure inter-domain operations.

10.2.1 Use of cryptography for communication security services

The use of cryptographic techniques contributes (to some extent) to the provision of all the security services required for inter-domain operations. It is possible to identify two main types of cryptographic algorithm, as described below.

- Symmetric (i.e. secret key) methods, in which knowledge of the encryption key implies knowledge of the decryption key and vice-versa. In order to preserve
confidentiality, the key must only be known by the sender and receiver. In addition, a non-repudiation service cannot be provided as it is impossible to prove which of the communicating parties had encrypted the data. The most common example of a symmetric algorithm is the Data Encryption Standard (DES) (National Bureau of Standards 1977).

![Symmetric Encryption Diagram](image)

**Fig. 10.1 : Symmetric Encryption**

- Asymmetric (i.e. public key) in which knowledge of the encryption key does not imply knowledge of the associated decryption key and it is computationally impossible to derive one from the other. The two keys are referred to as the “private” and “public” keys. Whilst the former remains a secret, the public key can be made available to all potential senders. Non-repudiation is, therefore, possible as only the legitimate owner should be able to encrypt messages with the private key (encryption in this manner is referred to as a digital signature). The most common asymmetric encryption method is based upon the RSA algorithm (Rivest et al. 1978).
Whilst the ability to provide an additional security service would seem to indicate that the public key approach is the more desirable of the two methods, a problem is that the encryption process is computationally intensive and, hence, much slower than the symmetric technique. This can be illustrated by comparing the speeds of two typical DES and RSA encryption chips (LINTEL 1992a; LINTEL 1992b). Whilst the DES processor is capable of encryption speeds in excess of 22 Mbit/sec, the RSA device can only manage a more sedate 32 Kbit/sec (using a 512 bit key). This performance constraint serves to make asymmetric methods unsuitable for encrypting large messages. The solution lies in the use of a hybrid system where the public key encryption is used to provide digital signature and integrity services and symmetric encryption is used for confidentiality. This serves to combine the "easy key exchange" of public key systems with the speed of symmetric algorithms.

Using a hybrid system, the required security services can be achieved as described below (note that in order to explain the concepts more clearly, the discussion will make reference to two communicating parties; sender S and recipient R).
Chapter 10: A Wider Framework for Healthcare Security

- Confidentiality

In order to use the faster symmetric encryption approach, the two parties wishing to communicate confidentiality both require access to the secret key. However, for confidentiality to be assured, it is obviously important that this key be known only to the communicating parties. Therefore, the party initiating the communication (i.e. S) would also initiate the generation of a key to be used for the duration of the exchange (i.e. a session key). This would then be sent to R, having first been encrypted using R’s public key (i.e. so that only R can read it). At the receiving end, R would use his secret key to decrypt the message, yielding the session key information to be used for the subsequent confidential transmission.

- Integrity

There are actually two possible techniques for ensuring message integrity. If the message is already being sent in encrypted form (i.e. for confidentiality), then this also provides an implicit integrity service, in that any modification of the encrypted data will result in garbage being generated at the receiving end.

Alternatively, integrity can be assured using a Message Authentication Code (MAC) which is appended to the message by the sender. The MAC is a hash function of the data itself, such that modification of the message would be highlighted by a subsequent discrepancy in the accompanying value. The recipient of the message would perform the same hash function on the data.
received to calculate his own MAC value for comparison (with non-matching values indicating that the message has been corrupted).

The MAC itself must obviously be protected to prevent someone from being able to modify the message and then substitute an appropriate new value. This is achieved by encrypting it with the public key of the recipient.

The use of the MAC approach has an advantage in that the process of code generation and comparison can be performed automatically (and transparently) at the receiving end. With the use of encryption alone, the recipient would still be required to manually view the message in order to determine whether it had been modified.

- **Non-repudiation**

  The sender S encrypts the message with the private key. R can then use the public key of S to decrypt the message. If the secrecy of the private key is assured, then whatever was signed with it could only have been sent by S and, therefore, the origin of the message cannot be repudiated. This digital signature can be used as a means of proving the source of session keys and MACs involved in the communication.

The hybrid approach, therefore, provides a basis for all of the necessary security services and the overall sequence for ensuring secure communications would be as follows (it is
assumed that the communicating parties are already in possession of the required public and private keys and that all three services are required:

1. a session key is generated locally by the sender (S);
2. the session key is transmitted to the receiver (R) after having been encrypted using the public key of R (for confidentiality) and signed using the secret key of S (for non-repudiation of origin);
3. S calculates a MAC for the message to be sent, which is again signed and encrypted (for non-repudiation of origin and confidentiality respectively), and appended to the message;
4. the message itself is encrypted using the shared session key (for confidentiality) and sent to R;
5. being in possession of the session key, R can decrypt the message;
6. using his own secret key and the public key of S, R can decrypt and verify the origin of the MAC;
7. finally, a new MAC value can be generated from the message received and compared against the original in order to ensure message integrity.

However, problems exist in this framework in terms of ensuring that:

- all potential senders and recipients are uniquely identifiable;
- public keys are securely associated with the correct user;
- public keys of users are available to other users when required.
These issues can be overcome by using a hierarchy of Trusted Third Parties (TTPs) to provide the required services, as found in X509 and ISO 9594-2 (ISO 1988).

10.2.2 An overview of Trusted Third Parties

The potential uses of TTPs in healthcare have already been recognised within Europe, having been the basis for a dedicated project under the INFOSEC programme (INFOSEC THIS 1994). This section identifies the key elements of a TTP service, highlighting the aspects of trust involved in each case. Summary descriptions are given below.

- **Naming Authority**

To ensure secure communications a unique and unforgeable identification of all potential users is necessary which can be bound to all activities or data used in a session.

The Naming Authority (NA) is responsible for assigning each communicating entity a *distinguished name* by which they may be identified within the communications framework (where such entities may be users, organisations or computer processes). Naming would actually be achieved via a hierarchy of NAs, arranged in an inverted tree structure as shown in figure 10.3. The entities at the bottom of the hierarchy represent individual users, with the root and intermediate levels all acting as NAs. Each entity is assigned a unique relative distinguished name by its superior NA and, thus, the overall distinguished name for any user is the concatenation of all relative names found along the closed path from the root.
Fig. 10.3: The Naming Authority hierarchy

The NA must be trusted that any name issued is unique and cannot be subsequently forged, changed or proliferated. A single person may possess more than one distinguished name (e.g. several issued by different NAs). In terms of the Directory (see below), one of these names will be held as a main entry and the rest as aliases.

- **Certification Authority**

The purpose of the Certification Authority (CA) is to provide assurance of user credentials (which will principally include their distinguished name and public encryption key) by producing a certificate which is then placed in the Directory. The CA signs the certificate with its own private key, ensuring that any user in possession of the associated public key can access it and that no-one can subsequently modify it without changes being detected. As such, certificates can be considered public and unforgable, and do not need to be communicated in a secure manner.
Users will sign data communications using their secret key. Recipients will require the associated public key and will, therefore, need to obtain the senders certificate. They can then use the public key of the CA to verify the certificate - yielding the senders public key which can be used to verify the integrity of the message and guarantee the source of the sender.

It is evident that CAs must be the highly secure, in that their compromise could lead to forged certificates. As such, the CAs also need to guarantee the authenticity of certificates to each other and a hierarchical arrangement is again used. This is illustrated in figure 10.4, along with the format of the certificates (with SK and PK representing the secret and public keys of the CAs at each stage).

The security of signing keys will also be of paramount importance to prevent the forging of user signatures. Storage of the key in a smart card is the commonly
recommended means of ensuring this (Rihaczek 1991; INFOSEC THIS 1994),
which will allow for transparency and ease of use by healthcare staff. The cards
perform various functions including the creation and verification of signatures,
encryption / decryption of data and the storage of secret keys or other sensitive data
(they may also perform other special functions associated with particular applications).

- **Directory**

  The directory is a distributed database, accessible on a potentially world-wide basis,
holding information on subscribers (principally distinguished names, aliases and
public keys) and provides an efficient means for public keys to be distributed to the
intended communications partners.

Access to directory entries is possible from any of a series of interconnected and
geographically dispersed directory service agents (DSAs), but with services
provided for user authentication. Whilst simple password methods could be used
for this purpose, stronger approaches (based on cryptographic techniques) are
considered more appropriate to healthcare (INFOSEC THIS 1994). Methods
should be of a comparable strength across all of the potential access points.

In addition, the directory must be trusted to maintain the integrity and, potentially,
the confidentiality (e.g. for the protection of alias names) of the information held.
10.2.3 Implementing a Trusted Third Party infrastructure

In order to provide all the necessary functions on an international scale a network of TTPs will be required, as shown in figure 10.5. At this level the infrastructure will be generic for all applications, but at the local domain and sub-domain levels (as shown in figure 10.6) specific operations can be incorporated to satisfy HCE security policies, with the TTP being linked into the more comprehensive SMC functionality described earlier.

![Diagram of TTP infrastructure at the international and national levels](image1)

![Diagram of TTP infrastructure at the national and local levels](image2)
Each TTP in the hierarchy is certified by the TTP in the next layer up, which not only provides credibility of the complete system by defining the individual certification path within a certificate, but allows for the loss of a hierarchical level under fault conditions (with the next higher order certificate being used). The arrangement is defined in the X509 / ISO 9594-2 Directory services architecture and helps to ensure that system failure does not lead to service unavailability.

The actual authorisation and supervision of inter-domain operations would be based upon interactions between the Security Management Centres involved. For example, to enable a user in domain A to utilise facilities on a system in a remote domain B, the relevant user details would be exchanged between the two SMCs. SMC A could (for example) send a signed behaviour profile to SMC B which, after ensuring that the user is actually authorised to access the system, could be used for subsequent session supervision.

The SMCs would attempt to harmonise the security services offered at each end. In the healthcare scenario, this goes some way towards satisfying a previously identified need for a mechanism whereby the level of security determined for data in one establishment can still be guaranteed to apply after transmission to another (AIM SEISMED 1993a).

With a secure association established, inter-domain operations can occur as normal, with the fact of the SMC communications remaining transparent to the end-user(s) involved. This would also be true of the other inter-domain services discussed, with all technical operations being handled by the security systems in each domain. However, in some cases, such as the
use of digital signatures, it would be advantageous for the users to be given some indication
that a security service is being provided (which could be accomplished by the fact of having
to use the smart cards).

10.3 Inter-HCE communication in an example scenario

The likely types of inter-domain communication in healthcare and the consequent need for
security can be illustrated using an example scenario. To this end, the information flows
involved in a potential system are illustrated in figure 10.7 and explained in the description
below, showing how both facilities and expertise can be shared via inter-domain
networking.

Fig. 10.7: Potential inter-domain HCE communications
Chapter 10: A Wider Framework for Healthcare Security

The neurology department in one establishment (HCE A) performs a series of tests which produce a set of "raw" results data. However, HCE A lacks the equipment required to process and visualise the data, making it necessary to involve facilities at another site (HCE B). Once visualisation has been performed the results are stored in a database, from where they are subsequently accessed by a consultant at HCE A. However, further expert opinion is required and advice is, therefore, sought from another neurological consultant located at HCE C. Hence, the data is transmitted further, with the additional interpretation finally coming back to the originating consultant (allowing a more informed care decision to be made at HCE A). Both consultants have access to a video conferencing link by means of camera-equipped workstations, whilst the other party at HCE A uses a standard workstation without such a facility.

From this basic outline, a general security specification can be given based upon elements of the strategy described earlier. The different HCEs would be authenticated to each other by the certified public keys obtained from the TTP, with all parties being authenticated locally by their respective SMCs. Given that their workstations are equipped with cameras, the two consultants could potentially be authenticated by an image recognition system. However, the data production user, utilising a standard workstation, would have no facility for multimedia-enhanced authentication methods. Authentication of this user would, therefore, be reliant upon the SMC facilities for activity supervision (possibly alongside traditional methods or the use of smart cards).
The example is heavily communications oriented and the SMCs in each domain would mediate the various data exchanges and messages. The latter would be protected according to the sequence of operations previously listed in section 10.2.1. The principal services required between HCEs A and B would be data integrity and confidentiality, whereas the HCE A / HCE C link would also require digital signatures so that the consultants were able to verify the origin of the messages received.

The same scenario, but with the SMCs and logical security information flows indicated, is shown in figure 10.8. The SMCs would monitor the activity in each of the three domains, with communication via secure protocols to exchange relevant security information (e.g. the behaviour profile of the data production user from SMC A to SMC B to allow supervision in the remote domain). They would also provide public keys for the verification of digital signatures.

The example serves to illustrate both the types of inter-domain information exchange and consultations that will be possible in healthcare, as well as the need for secure data communications between the various parties involved. The use of the SMCs would ensure that security was consistent across the three sites involved (a factor that considerably reduces the potential problems of sharing data and facilities as discussed), whilst the TTP certification hierarchy would ensure that SMCs could be authenticated to each other.
10.4 Conclusions

In conclusion, it can be seen that the adoption of this strategy would provide the final component of a logical security system necessary to allow secure inter-HCE operations. The combination of real-time supervision, the CISS framework and TTPs theoretically allows full system-wide protection to be realised, with trusted communication paths between a number of individually secure healthcare domains.
High level aspects of this framework have been published in Furnell et al (1995c) and Furnell and Sanders (1995). The former specifically examined the need for security in multimedia healthcare systems and suggested that the combination of user supervision within local domains and TTP-based communication between HCEs is a suitable strategy. The second paper concentrated more upon the TTP issues and the standardisation of healthcare security. Copies of these papers can again be found in appendix F.
CHAPTER 11

Conclusions
11.1 Achievements of the research programme

The research programme has met all of the objectives originally specified in chapter 1, with new conceptual and practical work being encompassed in a number of areas, as listed below.

1. Recommendation of the baseline standard for existing healthcare systems, in the form of the guidelines contributed to the AIM SEISMED project. These satisfy the basic requirement of providing a viable means for security to be added or enhanced in poorly protected systems and environments.

2. Development of the new protection methodology framework, which allows HCEs to assess their own security requirements and thereby determine the level of countermeasures that are necessary.

3. Development of the comprehensive conceptual design for the IMS supervision system. The framework specifies a novel combination of auditing, advanced authentication and supervision techniques that are considered suitable for use in healthcare systems. Several aspects of the design also represent entirely new work, including the class profiles for healthcare staff and the IMS module architecture.

4. Implementation and validation of a practical real-time user supervision technique, namely keystroke analysis. The experimental study involved a reasonably large test subject group and was, therefore, considered to provide a good indication of
the effectiveness of the technique. The results themselves proved that, in the vast
majority of cases, keystroke characteristics do indeed provide a suitable basis for
discriminating between many different users (even those with similar typing
abilities).

5. Implementation of various other key aspects from the conceptual design within a
demonstrator system. As well as incorporating further unique aspects of the IMS
design (e.g. system configuration auditing), this enabled the idea of transparent
supervision to be shown in practice, thereby also validating that part of the
concept.

6. Consideration of the need for security from a wider perspective and with a full
examination of how the recommended supervision approach could be integrated
into a more comprehensive framework (i.e. by using the CISS architecture and
Trusted Third Parties).

Several papers relating to the research programme have been presented at refereed
conferences, with favourable comments being received from other delegates. As such, it is
believed that the research has made valid and useful contributions to the information system
security field in both the healthcare context, as well as at a more general level.
11.2 Limitations of the research

Despite having met the overall objectives of the research programme, it is nevertheless possible to identify a number of limitations associated with the work. The principal points are presented below.

1. The generic protection methodology, whilst relatively complete in terms of the overall framework, cannot be considered practically viable due to the current lack of associated countermeasures. However, as noted in chapter 5, this problem will be overcome by efforts currently underway within another research programme (Warren 1995).

2. Insufficient time was available to allow the investigation of further supervision techniques in a practical context. Preliminary work was started relating to the potential of other behavioural characteristics (for example, analysis of operating system command usage), but did not progress significantly beyond the stage presented in chapter 6. It was considered more beneficial to devote time to a detailed treatment of keystroke analysis (and the associated demonstrator) rather than attempt several high level studies of other techniques.

3. It was not possible to conduct trials of the intrusion monitoring system within an operational HCE system, despite the fact that reference centre implementation of an IMS was originally one of the proposed goals of the SP07 workpackage within SEISMED. In the event, practical implementation work was deemed outside the scope of the workpackage by the project management team (AIM...
SEISMED 1993d). Nevertheless, significant interest in the real-time supervision concept was expressed by both of the UK reference centres.

11.3 Suggestions and scope for future work

It is possible to identify a number of areas in which further work could be conducted to build upon that undertaken within the project. A number of ideas have already been covered at the end of chapters 8 and 9, as well as throughout chapter 10. However, this section presents some additional points which the author views as representing a more direct continuation of the research programme.

1. Assessment of the applicability of the baseline security guidelines in practical trials, with associated updates to overcome any deficiencies. As previously noted in chapter 4, the guidelines would also require periodic updates to account for changes and advances in available protection technologies.

2. Further investigation and potential development an expert system implementation of the generic protection methodology, as identified in chapter 5. This would further enhance the ease of use and applicability of the approach once the accompanying countermeasures have been defined.

3. Enhancement of the keystroke analysis system to determine whether false acceptances can be further reduced using some of the recommendations made in chapter 8.
Chapter 11: Conclusions

4. Investigation and inclusion of further supervision characteristics within an operational IMS. The incorporation of artificial intelligence techniques and implementation in a full LAN environment (as discussed in chapter 9) would also provide a very interesting basis for future development.

5. Establishment of a more formal relationship between the IMS and CISS architectures, with the two concepts eventually being integrated into a functional system (initially in the context of an expanded demonstrator).

6. The aforementioned practical trials in a healthcare establishment.

11.4 The future for healthcare information systems and security

It is now unquestionable that information systems are firmly established in the healthcare field. However, in many ways the key issue is not so much the pervasiveness of IT, but the fact that the role it plays has changed so significantly. Systems have moved on from handling statistical and administrative duties to holding sensitive clinical information relating to individuals. In the future, more and more patient care will be conducted on the basis of information provided by IT systems, with further aspects of the treatment itself also being computer-controlled.

The future will see further growth and expansion of both applications and users, with the issues of multimedia and telemedicine playing an increasingly important role. In order to maintain confidence, security will need to be considered and developed in parallel with these
new systems, with each innovation being secure in itself and not compromising or degrading
the protection that already exists (Sunday Times 1995).

This research has served to highlight not only the need for security in these systems, but
also suggested a number of suitable strategies for actually addressing and overcoming some
of the main problems.

The significant potential for system abuse in healthcare has been established in this thesis.
This fact, in combination with the additional risks from accidental breaches, already means
that healthcare systems are under threat 24 hours a day. The increasing reliance upon and
use of IT systems in the future will simply mean that there is more to go wrong.

However, it is also expected that the available methods of security will advance and it is
envisaged that intrusion monitoring methods, such as those described in this thesis, will
become increasingly suitable as a means of providing transparent protection that does not
impose unwanted and unnecessary constraints upon users.

In conclusion, computer security will always be required in healthcare to safeguard both the
rights and safety of patients, as well as to allow the systems themselves to be used with
confidence. In fact, as the applications become more complex, the importance of having
adequate security can only increase.


5. AIM SEISMED. 1993c. "Data Sensitivity Survey Results", SP07-0.6. S.M.Furnell and P.W.Sanders, University of Plymouth, United Kingdom.


References


   Profiles”, In *SEISMED Training Programme : Guidelines for the Security of Health 
   Information* (Thessaloniki, Greece, Sept. 26-30).

   *Telematics for Health Care : Its impact ? Its future ?*. Produced by ACOSTA for the 
   Commission of the European Union AIM programme. (Dec.).

   typing biometrics”, In *Information Security - the Next Decade: Proceedings of the 
   IFIP TC11 eleventh international conference on information security, IFIP/Sec '95. 

   Software Engineering* 13, no.2: 222-232.


   5200.28-STD, United States of America (Dec.).


98. Morrissey, J.P. 1995. A hardware implementation of a local security unit for CISS. PhD research programme. School of Electronic, Communication and Electrical Engineering, University of Plymouth, United Kingdom.


105. NHS. 1986. *A National Framework for Information Management in the Hospital and Community Health Services*, Information Management Group (Oct.).


References

future of data and telecommunications management (Pretoria, South Africa, May 27-29).


References


APPENDIX A

Surveys associated with the research programme
Appendix A: Surveys associated with the research programme

As previously described in the main text, the results of two surveys were used to support the investigations in the research programme:

1. the SEISMED questionnaire (previously described in chapter 3), which was distributed to HCEs across Europe in 1992 and contained questions relevant to the assessment of existing systems security (pages 372 to 376);

2. the “Security Attitudes” questionnaire (previously described in chapter 6), which was distributed to staff within Derriford Hospital (Plymouth) in 1995 and contained questions relevant to the formulation of healthcare behaviour profiles (pages 378 to 382).

The questionnaire sheets associated with both surveys are reproduced in the pages that follow. However, given that the full SEISMED questionnaire was quite large (running to 23 pages), the appendix only reproduces those questions that related to the analysis of existing healthcare systems - namely those from sections one and two of the document.

For completeness, this section also presents a list of the major HCE applications that were identified under the heading “other” in question 1.3 of the SEISMED survey. This can be found directly after the SEISMED questionnaire pages themselves (i.e. page 377).
Appendix A: Surveys associated with the research programme

SECTION 1: PHYSICAL SECURITY DETAILS OF YOUR SYSTEM

1.1 Indicate in the appropriate boxes below your computer type, operating system, and whether or not they are networked.

<table>
<thead>
<tr>
<th>Computer Type</th>
<th>Operating System</th>
<th>Networked</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Manufacturer and Model No.)</td>
<td>(including version No.)</td>
<td>(Y/N)</td>
</tr>
<tr>
<td>Mainframe / Mini's</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Personal Computers</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1.2 Please give approximate answers to the following:

- No. of terminals?......................................................................... 1
- No. of users?............................................................................... 2
- Disk storage capacity? ...................................................... (Mbytes) 3

1.3 Indicate (Y/N) which of the Main Systems listed below are present within your HCE and which of the security measures are used?

<table>
<thead>
<tr>
<th>SECURITY MEASURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient index (demographic information)</td>
</tr>
<tr>
<td>Patient records (clinical data)</td>
</tr>
<tr>
<td>Clinical Laboratory</td>
</tr>
<tr>
<td>Pharmacy</td>
</tr>
<tr>
<td>Admissions</td>
</tr>
<tr>
<td>Financial Systems</td>
</tr>
<tr>
<td>Other (please name)</td>
</tr>
<tr>
<td>Other (please name)</td>
</tr>
<tr>
<td>Other (please name)</td>
</tr>
</tbody>
</table>

A = User authentication - identification
B = Audit trails - Intrusion detection mechanisms
C = Access control mechanisms
D = Physical security measure
E = Disaster - recovery plans
F = Encryption

372
### Appendix A: Surveys associated with the research programme

**1.4** How is User Authentication carried out in up to 5 of the Main Systems answered Y in Q.1.3? Enter the row numbers from Q.1.3 of these Main Systems in the 1st row of boxes and then tick the relevant boxes below that for each such system.

<table>
<thead>
<tr>
<th>Description</th>
<th>Box Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passwords</td>
<td>1</td>
</tr>
<tr>
<td>Group/Shared passwords</td>
<td>2</td>
</tr>
<tr>
<td>User name/password</td>
<td>3</td>
</tr>
<tr>
<td>System/user name/password</td>
<td>4</td>
</tr>
<tr>
<td>Challenge response exchange</td>
<td>5</td>
</tr>
<tr>
<td>No authentication</td>
<td>6</td>
</tr>
<tr>
<td>Other (please name)</td>
<td>7</td>
</tr>
<tr>
<td>Other (please name)</td>
<td>8</td>
</tr>
</tbody>
</table>

**1.5** What physical security measures are used (please tick)?

<table>
<thead>
<tr>
<th>Description</th>
<th>Box Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lock and key (Room)?</td>
<td>1</td>
</tr>
<tr>
<td>Swipe card entry (Room)?</td>
<td>2</td>
</tr>
<tr>
<td>Keypad entry (Room)?</td>
<td>3</td>
</tr>
<tr>
<td>Lock and key (Computer)?</td>
<td>4</td>
</tr>
<tr>
<td>Fire proof safe (Backups)?</td>
<td>5</td>
</tr>
<tr>
<td>None?</td>
<td>6</td>
</tr>
<tr>
<td>Other</td>
<td>7</td>
</tr>
</tbody>
</table>

Specify:

**1.6** Which of the following network type/configuration do you use (please tick)?

<table>
<thead>
<tr>
<th>Description</th>
<th>Box Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Area Network (LAN)?</td>
<td>1</td>
</tr>
<tr>
<td>Wide Area Network (WAN)?</td>
<td>2</td>
</tr>
<tr>
<td>X25?</td>
<td>3</td>
</tr>
<tr>
<td>Token Bus?</td>
<td>4</td>
</tr>
<tr>
<td>Token Ring?</td>
<td>5</td>
</tr>
<tr>
<td>Star?</td>
<td>6</td>
</tr>
<tr>
<td>Ethernet?</td>
<td>7</td>
</tr>
<tr>
<td>Other (Please name)?</td>
<td>8</td>
</tr>
<tr>
<td>Other (Please name)?</td>
<td>9</td>
</tr>
</tbody>
</table>
Appendix A: Surveys associated with the research programme

1.6.1 Which of the following type(s) of data base system(s) do you use (if any) (please tick)?

- Relational? ................................................................. 1
- Network/Hierarchical? .................................................. 2
- Other (Please name)? ................................................... 3
- On micro(s)? .............................................................. 4
- On mainframe(s)? ........................................................ 5
- Used for cooperative? .................................................... 6
- Local DB applications? .................................................. 7
- Integrated system? ....................................................... 8
- Distributed DB system? ................................................ 9

1.6.2 How is data base security controlled and maintained in your system (please tick)?

a) by physically separating subsystems .................................. 1
b) other (please name) ...................................................... 2

If the answer to 1.6.2 is (a), how do you achieve reintegration (please tick)?

- as part of a specific DB application ................................... 1
- by using a centralised DB as a generic application tool ........ 2
- by integrating at the network level .................................... 3
- other (please name) .................................................... 4

1.6.3 What overall Data Base security policy do you use (if the DB is seen as a technical tool for communication, eg. military, commercial, personal knowledge approaches etc)

(please name) ..................................................................

1.7 Please rank each of the following in terms of the frequency with which they causes problems (1 Often, 2 Occasionally, 3 Very occasionally, 4 Once ever, 5 Never)

- Unreliable hardware .................................................... 1
- Unreliable software ....................................................... 2
- Unreliable network ...................................................... 3
- Physical protection of hardware ....................................... 4
- Insufficient backups ................................................... 5
- Natural disaster .......................................................... 6
- Power loss .............................................................. 7
- Abuse by authorized users ............................................. 8
- Internal hacking ...................................................... 9
- External hacking ..................................................... 10
- Viruses ............................................................ 11
- Loss of data integrity ................................................. 12
- Other (please name) .................................................. 13
- Other (please name) .................................................. 14

1.8 For how many hours per day are your applications intended to be operational?

0-8 hrs, 8-12 hrs, 12-20 hrs, 20-23 hrs, 23-23.75 hrs, 23.75-24 hrs ............
Appendix A: Surveys associated with the research programme

SECTION 2: DESIGN SECURITY DETAILS OF YOUR SYSTEM

2.1 What kind of access control mechanisms do you use (please tick):
- Access control lists? ......................................................... 1
- Access rights to read or write data? ....................................... 2
- Captive accounts (No access to command line)? ....................... 3
- Alarms for access violations? .............................................. 4
- Enforced password changes? .............................................. 5
- Electronic mail? ................................................................. 6
- Automatic timed log out? (after how long?) ............................ 7
- Terminal locking when inactive? .......................................... 8
- Configuration management? .............................................. 9
- Other (please name)? ...................................................... 10
- Other (please name)? ...................................................... 11

Please indicate the name and telephone number of someone who would be willing to further discuss the security within your Hospital Systems.

2.2 Do you have any of the following disaster-recovery measures (Y/N) in your system:
- Backups (state type eg. weekly/daily) .................................. 1
- Duplicate computer system? .............................................. 2
- 24-hour maintenance contract? ........................................... 3
- Mobile backup? ................................................................... 4
- Other (please name) ......................................................... 5

2.3 Are your disaster-recovery procedures designed to cope with any of the following (Y/N):
- All systems recovery? ....................................................... 1
- Core systems? ................................................................. 2
- Network? ....................................................................... 3
- Personal computers? ....................................................... 4
- No disaster recovery procedures? ....................................... 5
- Other (please name) ......................................................... 6

2.4 Does your organisation use any form of encryption (Y/N):
- of data in systems? ......................................................... 1
- of messages on internal networks? ....................................... 2
- of messages on external networks? ...................................... 3
- of password files? ........................................................... 4
Appendix A: Surveys associated with the research programme

2.5 How is Data Integrity maintained in up to 5 of the Main Systems answered Y in Q.13. Enter the row numbers from Q.13 of these Main Systems in the 1st row of boxes and then tick the relevant boxes below that for each such system.

- Double entry of data .......... 1
- Non manual data entry .......... 2
- Error detection and correction software ........................................ 3
- Batch totals on data entry ...... 4
- Data validation on data entry . 5
- Selective sampling .............. 6
- On screen re-reading ............ 7
- No integrity checks ............ 8
- Other (please name) 9
- Other (please name) 10

2.6 How is access controlled and maintained across the network (please tick)?

- Automatic dialback modems .......................................................... 1
- Controlled access points ............................................................... 2
- Terminal authentication ................................................................. 3
- Node authentication ...................................................................... 4
- Encryption (state type) ................................................................. 5
- Secured cabling .............................................................. 6
- Wiretap detection ........................................................................ 7
- No restrictions ............................................................................ 8
- Other (please name) ................................................................. 9
- Other (please name) .................................................................10

2.7 What is the acceptable delay for information to be delivered by the HCE in:

1. Average cases? ........................................................................ 1
2. Emergency cases? .......................................................................

Use the following code:
1 = less than a second
2 = 1 to 5 seconds
3 = 5 to 10 seconds
4 = More than 10 seconds
Other European healthcare applications

The following is a list of 25 additional types of HCE application that were identified in the SEISMED survey and subsequently presented under the heading "other" in figure 3.1. The names given are as specified by the survey respondents themselves:

1. Personnel System;
2. Management Information System;
3. National indicators;
4. Waiting Lists;
5. Administration;
6. Births, deaths;
7. Spreadsheet;
8. Nursing;
9. Purchasing;
10. Contractor details;
11. Donor Records;
12. Radiology;
13. Cancer registration;
14. Child health;
15. Human Resources;
16. Materials Management;
17. Out-patients;
18. Meal supply;
19. Contractor details;
20. ECG system;
21. CRAMM risk analysis reviews;
22. Paramedic;
23. Surgery Audit;
24. Operation history;
25. Abortion register.
Appendix A: Surveys associated with the research programme

Security Awareness Survey

Thank you for taking the time to fill in this questionnaire, which is completely anonymous.

It is intended to be completed by information system users, so that we may gain a better understanding of your attitude and awareness of security relating to the systems you use. If you answer the questionnaire in full it will give an accurate picture of your opinions on this subject and allow proper analysis and suggest possible improvements.

The majority of the questions require a simple “yes” or “no” response, in which you should just tick the appropriate box for your answer. For the other questions, the style of response will be shown.

In completing the questionnaire it should be recognised that information security relates to the maintenance of the following concepts:

- Confidentiality: the requirement that information about someone or something can only be accessed by authorised persons.

- Integrity: the requirement that whatever you have stored on the computer system will still be the same when you come back to it.

- Availability: the requirement that systems and/or data will be always accessible to any user with a legitimate need.

Your help in completing this questionnaire is greatly appreciated.
General Information

1. Please tick the appropriate boxes:
   Sex: □ Male □ Female
   Age: □ Under 20 □ 20 - 29 □ 30 - 39
   □ 40 - 49 □ 50 - 59 □ 60 and over

2. Which of the following categories best describes your role?
   □ Consultant □ Junior Doctor □ Nurse
   □ Researcher □ Administrator □ Secretary
   □ PAM □ Clerk

3. What are your typical hours of work (if fixed)? From ___ to ___

4. On average, how long do you spend using the hospital computer systems
   _____ hours each day?

5. Please indicate the types of computer system that you regularly use?
   □ Standalone PC □ Terminal to Hospital Computer
   □ PC on a Network □ Remote (non-Derriford) System
   □ Other (please specify) ____________________

6. Please indicate which of the following types of applications you use and
   how frequently (1 whole of day, 2 part of day, 3 less frequently, leave
   blank if never used)
   □ PAS □ Clinical Workstation □ Clinical Laboratory
   □ Radiology □ Financial Systems □ Theatres
   □ Other (please specify) ____________________

7. Which of the following types of data do you create (C), access (A), update (U)? Please tick all boxes that apply:
   C A U
   □ □ □ Patient Care/Diagnosis
   □ □ □ Patient Administration
   □ □ □ Personnel
   □ □ □ Resource Management
   □ □ □ General Hospital Administration
   □ □ □ Financial
   □ □ □ Laboratory, Radiology or other service dept.

8. Would you be able to continue your work if unable to use the hospital
   computer systems for 3 to 4 hours? □ Yes □ No
Appendix A: Surveys associated with the research programme

9. Are you aware of the possible consequences of passing on confidential information? □ Yes □ No

10. If you needed help, do you know how to contact the main computer support staff? □ Yes □ No

11. Do you normally access information systems from more than one workstation/terminal? □ Yes □ No
   If yes, are these workstation/terminals in different areas of the Hospital? □ Yes □ No

12. What do you think information system security protects against?
   On a scale of 1 to 5. (1 = Least Important, 5 = Most Important)
   □ Fraud (e.g. Financial loss)
   □ Misuse (e.g. Personal work or games)
   □ Patient safety (e.g. Accurate laboratory results)
   □ Patient Confidentiality (e.g. Value of information)
   □ Other reason (please state)

13. Do you consider the present information system security restricts you in your work? □ Yes □ No
   If yes, how?

14. How do you feel about the information system security at this Hospital and the controls and processes that are in place at present? (Scale of 1 Unhappy to 5 Very Confident, please tick)
   Logical 1 □ 2 □ 3 □ 4 □ 5 □
   Physical 1 □ 2 □ 3 □ 4 □ 5 □
   Personnel 1 □ 2 □ 3 □ 4 □ 5 □

   **Physical security**

15. Do you always wear your ID badge at work? □ Yes □ No

16. Would you challenge someone not wearing an ID badge who was gaining information? □ Yes □ No

17. Are any areas of the Hospital monitored/under surveillance? □ Yes □ No
Appendix A: Surveys associated with the research programme

18. How do you dispose of sensitive computer data such as printouts, tapes and disks?
   □ Shred  □ Green Bag  □ General waste
   □ Other, please state ____________________________

Computer system security

19. Do you know of anyone who has breached the computer system security? □ Yes □ No

20. Do you know of anyone who has taken information off the system that they should not have? □ Yes □ No
    If yes how? □ Disk  □ Print

21. Are you aware of any controls to stop people deleting or changing information they should not? □ Yes □ No

22. Have you ever left your workstation/terminal logged on and unsupervised? □ Yes □ No

23. Have you ever used a workstation/terminal when logged in on someone else's password? □ Yes □ No

24. Do you legitimately share a group password? □ Yes □ No

25. Other than for legitimate purposes:
   (a) Does anyone else know your password? □ Yes □ No
   (b) Do you know other people's passwords? □ Yes □ No

26. Could someone guess your password (e.g. is it related to your name, car, hobbies, or a dictionary word)? □ Yes □ No

27. Have you ever kept a written record of your password? □ Yes □ No

28. Do you have to change your password? □ Yes □ No
    If yes, how often? Every ________ days

29. How often do you feel is a reasonable length of time between password changes? Every ________ days.
Appendix A: Surveys associated with the research programme

Personnel issues

30. Do you know of any clauses in your contract of employment regarding the security and use of information?  
☐ Yes  ☐ No

31. Do you know if the data protection act applies to you or information that you use?  
☐ Yes  ☐ No

32. Does any local information system security documentation exist in your Hospital?  
☐ Yes  ☐ No  
If no, is there a formal information system security document for users.  
☐ Yes  ☐ No

33. Are you held personally responsible for certain data? (i.e. are you the data owner)  
☐ Yes  ☐ No

34. Do you promote information system security to your subordinates?  
☐ Yes  ☐ No

35. Does your department ever review the security process in order to improve information system security?  
☐ Yes  ☐ No

36. Have you been given information system security training?  
☐ Yes  ☐ No  
If yes, by whom? _______________________

37. Are you given regular security awareness information?  
☐ Yes  ☐ No  
If yes, by whom? _______________________

Thank you for completing this questionnaire
APPENDIX B

Security Guidelines for Existing Healthcare Systems
Appendix B : Security Guidelines for Existing Healthcare Systems

The tables on the following pages present a comprehensive list of the AIM SEISMED guidelines for existing healthcare systems that were developed as part of this research programme and described in chapter 4.

Each table corresponds to one of the ten protection principles that were identified and the titles of the underlying guidelines are listed in each case, along with an indication of their perceived applicability to the different categories of HCE staff - i.e. General (G), Management (M) and IT & Security Personnel (IS).
## Appendix B: Security Guidelines for Existing Healthcare Systems

### ESP0100 Security Policy & Administration

<table>
<thead>
<tr>
<th>ESG0101</th>
<th>Security policy document</th>
<th>✓</th>
<th>✓</th>
<th>✓</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0102</td>
<td>Security co-ordination</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0103</td>
<td>Allocation of responsibility</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0104</td>
<td>Information security advice</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0105</td>
<td>Management planning</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

### ESP0200 Physical & Environmental Security

<table>
<thead>
<tr>
<th>ESG0201</th>
<th>Physical security perimeters</th>
<th>✓</th>
<th>✓</th>
<th>✓</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0202</td>
<td>Physical access controls</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0203</td>
<td>Procedural access controls</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0204</td>
<td>Sensitive areas</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0205</td>
<td>Location of equipment &amp; assets</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0206</td>
<td>Asset inventory</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0207</td>
<td>Movement &amp; removal of HCE property</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0208</td>
<td>Equipment security off-site</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0209</td>
<td>Theft &amp; damage prevention</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0210</td>
<td>Locks &amp; code combinations</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0211</td>
<td>Equipment procurement &amp; maintenance</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0212</td>
<td>Equipment disposal</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0213</td>
<td>Clear desk policy</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0214</td>
<td>Security of unattended entrances</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0215</td>
<td>Protection of cabling</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0216</td>
<td>Protection from fire</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0217</td>
<td>Fire safety awareness</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0218</td>
<td>Protection from water damage</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0219</td>
<td>Lightning protection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0220</td>
<td>Power supply</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0221</td>
<td>Air conditioning</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0222</td>
<td>Safety inspections</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

### ESP0300 Disaster Planning & Recovery

<table>
<thead>
<tr>
<th>ESG0301</th>
<th>Continuity planning</th>
<th>✓</th>
<th>✓</th>
<th>✓</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0302</td>
<td>Continuity plan development</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0303</td>
<td>Continuity plan testing</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0304</td>
<td>Continuity plan review &amp; update</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0305</td>
<td>Fallback procedures</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0306</td>
<td>Post-disaster access control</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0307</td>
<td>Public relations procedures</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

---

385
### Appendix B: Security Guidelines for Existing Healthcare Systems

#### ESP0400 Personnel Security

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0401</td>
<td>Recruitment procedures</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0402</td>
<td>Confidentiality agreement</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0403</td>
<td>Job practices</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0404</td>
<td>Staff appraisal</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0405</td>
<td>Trusted personnel</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0406</td>
<td>Employment termination</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0407</td>
<td>Use of external personnel</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0408</td>
<td>Security variance reporting</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

#### ESP0500 Training & Awareness

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0501</td>
<td>Job training</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0502</td>
<td>Use of systems &amp; applications</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0503</td>
<td>HCE training programmes</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0504</td>
<td>Specialist training courses</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0505</td>
<td>Awareness of specific issues</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0506</td>
<td>Training responsibilities</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Appendix B: Security Guidelines for Existing Healthcare Systems

<table>
<thead>
<tr>
<th>ESP0600</th>
<th>Information Technology Facilities Management</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0601</td>
<td>System Planning &amp; Control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0602</td>
<td>Configuration management</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0603</td>
<td>Operational change control</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0604</td>
<td>Capacity planning</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0605</td>
<td>External management of facilities</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0606</td>
<td>Testing of security features</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0607</td>
<td>Consistent security measures</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0608</td>
<td>Back-up procedure</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0609</td>
<td>Back-up retention &amp; storage</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0610</td>
<td>Back-up administration</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0611</td>
<td>Recovery from back-up</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0612</td>
<td>Media disposal</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0613</td>
<td>Inspection of media sent and received</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0614</td>
<td>Handling sensitive media</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0615</td>
<td>Security classification labelling</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0616</td>
<td>Control of removable media</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0617</td>
<td>Security of media in transit</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0618</td>
<td>Media disposal</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0619</td>
<td>Mandatory auditing of activity</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0620</td>
<td>Audit trail inspection</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0621</td>
<td>Restricted audit trail access</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0622</td>
<td>Operator log maintenance</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0623</td>
<td>Security variance management</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0624</td>
<td>Anti-virus policy</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0625</td>
<td>Virus prevention</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0626</td>
<td>Virus detection</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0627</td>
<td>Virus containment</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0628</td>
<td>Virus recovery</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0629</td>
<td>Virus awareness</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0630</td>
<td>Documentation availability</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0631</td>
<td>Documented operating procedures</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0632</td>
<td>Documentation of operating system features</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0633</td>
<td>Security of system documentation</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
### Appendix B: Security Guidelines for Existing Healthcare Systems

<table>
<thead>
<tr>
<th>ESP0700</th>
<th>Authentication &amp; Access Control</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0701</td>
<td>User identification</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0702</td>
<td>User authentication</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0703</td>
<td>Secure login procedure</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0704</td>
<td>Suppression of information</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0705</td>
<td>Account integrity messages</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0706</td>
<td>Passwords</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0707</td>
<td>Password secrecy</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0708</td>
<td>Password selection</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0709</td>
<td>Password monitoring</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0710</td>
<td>Password changes</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0711</td>
<td>Password issuing</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0712</td>
<td>Encrypted password storage</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0713</td>
<td>System management authentication</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0714</td>
<td>Least privilege operation</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0715</td>
<td>Documented access control policy</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0716</td>
<td>Object access restrictions</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0717</td>
<td>Enforced access paths</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0718</td>
<td>Unauthorised access attempts</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0719</td>
<td>Resource usage limits</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0720</td>
<td>Timed terminal lock-out</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0721</td>
<td>Review of access rights</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0722</td>
<td>Restriction of temporary staff access</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0723</td>
<td>Restriction of third party access</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0724</td>
<td>User-defined file protection</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESG0725</td>
<td>Sensitive system isolation</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0726</td>
<td>Control of system utilities</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0727</td>
<td>Prohibition of compromised controls</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0728</td>
<td>Implementation of manufacturer security patches</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
Appendix B: Security Guidelines for Existing Healthcare Systems

<table>
<thead>
<tr>
<th>ESP0800</th>
<th>Database Security</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0801</td>
<td>Demarcation</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0802</td>
<td>New systems</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0803</td>
<td>Operational versions</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0804</td>
<td>Back up</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0805</td>
<td>Change log</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0806</td>
<td>Database software specifications</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0807</td>
<td>Database software testing</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0808</td>
<td>Database software maintenance</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0809</td>
<td>Corrective maintenance</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0810</td>
<td>Adaptive and enhancing maintenance</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0811</td>
<td>Copies</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0812</td>
<td>Scope of security</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0813</td>
<td>Database security policy</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0814</td>
<td>Database administration</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0815</td>
<td>Access control</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0816</td>
<td>People</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0817</td>
<td>Education and awareness</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG0818</td>
<td>User authentication</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0819</td>
<td>Auditability</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0820</td>
<td>Inference control</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0821</td>
<td>Database control</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESP0900</th>
<th>System Maintenance</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG0901</td>
<td>Source program access control</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0902</td>
<td>Controlled modification</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0903</td>
<td>Changes to vendor software</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG0904</td>
<td>Testing and acceptance</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>ESG0905</td>
<td>Review of operating system changes</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESP1000</th>
<th>Legislation Compliance</th>
<th>G</th>
<th>M</th>
<th>IS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESG1001</td>
<td>Data Protection legislation</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG1002</td>
<td>Information system misuse</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG1003</td>
<td>Copyright software</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ESG1004</td>
<td>HCE security policy</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ESG1005</td>
<td>Protection of business records</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
APPENDIX C

Data Mappings of Operational Healthcare Systems
Appendix C: Data Mappings of Operational Healthcare Systems

The following pages present a series of data mappings to further illustrate how the data requirements of genuine operational healthcare systems can be encompassed by the generic data model that was described in chapter 5. The systems shown are from HCEs in Plymouth and Thessaloniki who collaborated with the SEISMED project, and the following applications are represented:

- Radiology (Plymouth);
- Mental Health (Plymouth);
- Staffing (Thessaloniki);
- Accounting (Thessaloniki);
- Pharmacy (Thessaloniki);
- Patient Administration (Thessaloniki).

The data requirements of the Thessaloniki Patient Administration system can be contrasted with those of the Plymouth equivalent that was presented in chapter 5.
Appendix C: Data Mappings of Operational Healthcare Systems
Appendix C: Data Mappings of Operational Healthcare Systems

Patient Identification
- Demographic Information
- Social Data

Patient Administration
- Waiting List info
- Theatre mgmt
- Ward and Bed mgmt
- Transport requirements
- Referral Details
- Discharge Details
- Portering

Patient Care
- Case history
- Diagnosis
- Care Plan
- Processes carried out
- Outcome

Accounting
- System (Thessaloniki)

Clinical Services
- Radiology info
- Pharmacy info
- Laboratory info

Finance
- Contracts
- Payroll
- Invoicing
- Purchasing
- Budgets

Hotel Services
- Estates Management
- Hospital supplies
- Catering
- Domestic/ cleaning
- Works data

Staff
- Personnel
- Rostering

Library and Information Services
- Medical Knowledge
- Drug Information
- Codings & Classifications
- Protocols
- Research

Expert Systems
- Decision Support
- Neural Networks

Communications Services
- Messages
  - Work Orders
  - Forms
  - Results

Resource Mgmt and planning
- Statistical summaries
- Planning info
- Future activity info

External Systems

395
Appendix C: Data Mappings of Operational Healthcare Systems

- **Patient Identification**
  - Demographic Information
  - Social Data

- **Patient Administration**
  - Waiting List info
  - Theatre mgmt
  - Ward and Bed mgmt
  - Transport requirements
  - Referral Details
  - Discharge Details
  - Portering

- **Patient Care**
  - Case history
  - Diagnosis
  - Care Plan
  - Processes carried out
  - Outcome

- **Pharmacy System (Thessaloniki)**

- **Clinical Services**
  - Radiology info
  - Pharmacy info
  - Laboratory info

- **Finance**
  - Contracts
  - Payroll
  - Invoicing
  - Purchasing
  - Budgets

- **Hotel Services**
  - Estates Management
  - Hospital supplies
  - Catering
  - Domestic/cleaning
  - Works data

- **Resource Mgmt and planning**
  - Statistical summaries
  - Planning info
  - Future activity info

- **Library and Information Services**
  - Medical Knowledge
  - Drug Information
  - Definitions
  - Coding & Classifications
  - Protocols
  - Research

- **Expert Systems**
  - Decision Support
  - Neural Networks

- **Staff**
  - Personnel
  - Rostering

- **Communications Services**
  - Messages
    - Work Orders
    - Forms
    - Results

- **External Systems**

396
Appendix C: Data Mappings of Operational Healthcare Systems

Patient Identification
- Demographic Information
- Social Data

Patient Administration
- Waiting List info
- Theatre mgmt
- Ward and Bed mgmt
- Transport requirements
- Referral Details
- Discharge Details
- Portering

Patient Care
- Case history
- Diagnosis
- Care Plan
- Processes carried out
- Outcome

Clinical Services
- Radiology info
- Pharmacy info
- Laboratory info

Finance
- Contracts
- Payroll
- Invoicing
- Purchasing
- Budgets

Hotel Services
- Estates Management
- Hospital supplies
- Catering
- Domestic/cleaning
- Works data

Staff
- Personnel
- Rostering

Resource Mgmt and planning
- Statistical summaries
- Planning info
- Future activity info

Library and Information Services
- Medical Knowledge
- Drug Information
- Definitions
- Codings & Classifications
- Protocols
- Research

Expert Systems
- Decision Support
- Neural Networks

Communications Services
- Messages
  - Work Orders
  - Forms
  - Results

External Systems
APPENDIX D

Text Samples from the Keystroke Analysis Study
Appendix D: Text Samples from the Keystroke Analysis Study

The following pages present details of the three text samples that were used in the experimental study of keystroke analysis described in chapter 8. These texts were used to provide:

- the reference text; samples of which were used to create the user typing profiles;

- the two test sample texts; samples of which were used to determine authentication thresholds for legitimate subjects and represent impostor typing attempts.

In each case the full text passage is reproduced, along with an associated table listing the frequency of characters and character digraphs within it. This analysis is particularly relevant in the case of the reference text, as it shows the significant representation of the most common English language digraphs (which is important in relation to the issue of text composition that was discussed in section 8.3.3.1).

All of the texts were taken from existing literature and the source is cited in each case. It should be noted the text composition criteria were only applied in the case of the reference text. The other texts were simply required to be of a sufficient length to allow a reasonable typing sample to be obtained (thus allowing somewhat more arbitrary selection).
DIFFERENT PEOPLE HAVE DIFFERENT OBJECTIVES IN THEIR QUEST FOR FITNESS. TO AN ATHLETE ITS SECONDS SHAVED OFF A MILE OR THAT EXTRA BURST OF SPEED IN THE LAST MINUTE OF THE GAME. TO A LAWYER ITS ALERTNESS AFTER HOURS OF HARD BARGAINING. TO A HOUSEWIFE A DRESS TWO SIZES SMALLER OR MAYBE JUST THE SENSE OF ABUNDANT WELL BEING THE POSITIVE OUTLOOK AND REGAINED YOUTHFULNESS THAT COMES FROM BEING REALLY FIT. NO MATTER WHAT YOUR PARTICULAR EXERCISE AIM MAY BE THE MOST IMPORTANT THING IS TO ACHIEVE IT SAFELY. AFTER ALL YOU WANT TO GAIN YOUR HEALTH NOT LOSE IT. THATS WHY A THOROUGH PHYSICAL EXAMINATION SHOULD BE THE VERY FIRST STEP ON YOUR ROAD TO FITNESS. EMPHASIZING THE IMPORTANCE OF SUCH AN EXAMINATION IS THE FOLLOWING TRAGIC INCIDENT. ON JULY 22 A LEADING WEST COAST NEWSPAPER EXPLODED A BANNER HEADLINE : TWO MORE JOGGERS ! OTHER NEWSPAPERS ACROSS THE COUNTRY PICKED UP THE STORY. OCCURRING SHORTLY AFTER A WAVE OF ENTHUSIASM HAD MADE JOGGING SOMETHING OF A NATIONWIDE SPORT THE TRAGEDY SUDDENLY FOCUSED NATIONAL CONCERN ON THE PROBLEM OF SAFETY IN EXERCISE. MY PHONE RANG ALMOST CONSTANTLY. PHYSICIANS AND LAW PEOPLE ALIKE WERE ANXIOUSLY ASKING UNDER WHAT CONDITIONS EXERCISE MIGHT BE DANGEROUS AND PROMINENT DOCTORS WERE WONDERING OUT LOUD IN NEWSPAPERS AND MAGAZINES WHETHER PERHAPS THE IDEA OF EXERCISE HAD BEEN OVERSOLD TO THE PUBLIC. IN RESPONSE TO THIS WIDESPREAD CONCERN I DECIDED TO INVESTIGATE MORE CLOSELY THE CASE OF THE TWO JOGGERS WHO HAD SUFFERED FATAL HEART ATTACKS DURING THE EXERCISE. BOTH IT TURNED OUT HAD SEVERE HEART DISEASE AND ONE OF THEM HAD BEEN TOLD BY A PROMINENT WEST COAST PHYSICIAN THAT HE SHOULD UNDER NO CIRCUMSTANCES ENGAGE IN VIGOROUS EXERCISE. YET CONTRARY TO MEDICAL ADVICE THE MAN STARTING JOGGING AT A STRENUOUS RATE MISTAKENING BELIEVING THAT THIS WOULD HELP HIM OVERCOME HIS HEART CONDITION MORE QUICKLY. INSTEAD JOGGING AT A HARD PACE STRAINED HIS WEAK HEART BEYOND ITS LIMITS. THE ONE GOOD THING GROWING OUT OF THIS TRAGEDY WAS THE REALIZATION ON THE PART OF PHYSICIANS THAT ANYONE ENTERING AN EXERCISE OR PHYSICAL CONDITIONING PROGRAM SHOULD HAVE A MEDICAL CHECK UP BEFORE STARTING. SO BEFORE YOU EMBARK ON ANY EXERCISE PROGRAM GET YOUR DOCTORS APPROVAL.


Sample Text 1 (574 characters)

JIM GARRISON SERVED AS DISTRICT ATTORNEY OF NEW ORLEANS FOR TWELVE YEARS. THREE YEARS AFTER PRESIDENT KENNEDYS MURDER IN DALLAS IN NOVEMBER NINETEEN SIXTY THREE HE HEADED THE INTENSIVE INQUIRY THAT YIELDED THE ONLY CRIMINAL PROSECUTION EVER BROUGHT IN THE KENNEDY CASE. IT WAS THE BEGINNING OF A RELENTLESS SEARCH FOR THE TRUTH, A TRUTH SO SHOCKING THAT MANY PEOPLE WILL BE UNWILLING TO ACCEPT IT. IT IS A HARROWING ACCOUNT OF HOW THE AMERICAN GOVERNMENT AND THE MEDIA SPENT TWENTY FIVE YEARS TRYING TO GET THE WORLD TO BELIEVE A FAIRY TALE AND HOW THEY VERY NEARLY SUCCEEDED


Sample Text 2 (389 characters)

FOUR OUT OF FIVE ORGANISATIONS ARE NOW BEING HIT BY IT SECURITY BREACHES WHICH COST BUSINESS 1.2 BILLION A YEAR. THE STATISTICS COME FROM A NEW SURVEY BY THE DEPARTMENT OF TRADE AND INDUSTRY, ICL AND THE NATIONAL COMPUTER CENTRE. THE LOSSES ARE UP ON A SIMILAR SURVEY TWO YEARS AGO. IT SHOWS THAT FIRE WAS THE MOST FEARED IT SECURITY THREAT, WITH EQUIPMENT FAILURE THE COMMONEST PROBLEM.

## Reference Text

Appendix D: Text Samples from the Keystroke Analysis Study

### Unigram Frequency

<table>
<thead>
<tr>
<th>Rank</th>
<th>Unigraph</th>
<th>Freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E</td>
<td>368</td>
</tr>
<tr>
<td>2</td>
<td>T</td>
<td>223</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>162</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>151</td>
</tr>
<tr>
<td>5</td>
<td>O</td>
<td>147</td>
</tr>
<tr>
<td>6</td>
<td>N</td>
<td>130</td>
</tr>
<tr>
<td>7</td>
<td>I</td>
<td>129</td>
</tr>
<tr>
<td>8</td>
<td>S</td>
<td>121</td>
</tr>
<tr>
<td>9</td>
<td>R</td>
<td>113</td>
</tr>
<tr>
<td>10</td>
<td>H</td>
<td>88</td>
</tr>
<tr>
<td>11</td>
<td>D</td>
<td>67</td>
</tr>
<tr>
<td>12</td>
<td>L</td>
<td>57</td>
</tr>
<tr>
<td>13</td>
<td>C</td>
<td>57</td>
</tr>
<tr>
<td>14</td>
<td>G</td>
<td>54</td>
</tr>
<tr>
<td>15</td>
<td>U</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>P</td>
<td>43</td>
</tr>
<tr>
<td>17</td>
<td>M</td>
<td>41</td>
</tr>
<tr>
<td>18</td>
<td>F</td>
<td>39</td>
</tr>
<tr>
<td>19</td>
<td>Y</td>
<td>38</td>
</tr>
<tr>
<td>20</td>
<td>W</td>
<td>29</td>
</tr>
<tr>
<td>21</td>
<td>B</td>
<td>22</td>
</tr>
<tr>
<td>22</td>
<td>V</td>
<td>16</td>
</tr>
<tr>
<td>23</td>
<td>X</td>
<td>13</td>
</tr>
<tr>
<td>24</td>
<td>K</td>
<td>10</td>
</tr>
<tr>
<td>25</td>
<td>J</td>
<td>8</td>
</tr>
<tr>
<td>26</td>
<td>Z</td>
<td>4</td>
</tr>
<tr>
<td>27</td>
<td>Q</td>
<td>2</td>
</tr>
<tr>
<td>28</td>
<td>R</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>OR</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>E</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>M</td>
<td>1</td>
</tr>
<tr>
<td>32</td>
<td>IS</td>
<td>1</td>
</tr>
<tr>
<td>33</td>
<td>W</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>O</td>
<td>1</td>
</tr>
<tr>
<td>35</td>
<td>Y</td>
<td>1</td>
</tr>
<tr>
<td>36</td>
<td>B</td>
<td>1</td>
</tr>
<tr>
<td>37</td>
<td>ES</td>
<td>1</td>
</tr>
<tr>
<td>38</td>
<td>C</td>
<td>1</td>
</tr>
<tr>
<td>39</td>
<td>AL</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>A</td>
<td>1</td>
</tr>
<tr>
<td>41</td>
<td>EA</td>
<td>1</td>
</tr>
<tr>
<td>42</td>
<td>ED</td>
<td>1</td>
</tr>
<tr>
<td>43</td>
<td>ND</td>
<td>1</td>
</tr>
<tr>
<td>44</td>
<td>OF</td>
<td>1</td>
</tr>
<tr>
<td>45</td>
<td>CI</td>
<td>1</td>
</tr>
<tr>
<td>46</td>
<td>TI</td>
<td>1</td>
</tr>
<tr>
<td>47</td>
<td>EN</td>
<td>1</td>
</tr>
<tr>
<td>48</td>
<td>TO</td>
<td>1</td>
</tr>
<tr>
<td>49</td>
<td>AR</td>
<td>1</td>
</tr>
<tr>
<td>50</td>
<td>IT</td>
<td>1</td>
</tr>
<tr>
<td>51</td>
<td>F</td>
<td>1</td>
</tr>
<tr>
<td>52</td>
<td>IC</td>
<td>1</td>
</tr>
<tr>
<td>53</td>
<td>CO</td>
<td>1</td>
</tr>
<tr>
<td>54</td>
<td>RO</td>
<td>1</td>
</tr>
<tr>
<td>55</td>
<td>NT</td>
<td>1</td>
</tr>
<tr>
<td>56</td>
<td>RT</td>
<td>1</td>
</tr>
<tr>
<td>57</td>
<td>AD</td>
<td>1</td>
</tr>
<tr>
<td>58</td>
<td>BE</td>
<td>1</td>
</tr>
<tr>
<td>59</td>
<td>DE</td>
<td>1</td>
</tr>
<tr>
<td>60</td>
<td>VE</td>
<td>1</td>
</tr>
<tr>
<td>61</td>
<td>EX</td>
<td>1</td>
</tr>
<tr>
<td>62</td>
<td>TE</td>
<td>1</td>
</tr>
<tr>
<td>63</td>
<td>RA</td>
<td>1</td>
</tr>
<tr>
<td>64</td>
<td>RC</td>
<td>1</td>
</tr>
<tr>
<td>65</td>
<td>HI</td>
<td>1</td>
</tr>
<tr>
<td>66</td>
<td>RS</td>
<td>1</td>
</tr>
<tr>
<td>67</td>
<td>TA</td>
<td>1</td>
</tr>
<tr>
<td>68</td>
<td>F</td>
<td>1</td>
</tr>
<tr>
<td>69</td>
<td>DI</td>
<td>1</td>
</tr>
<tr>
<td>70</td>
<td>MI</td>
<td>1</td>
</tr>
</tbody>
</table>

### Digram Frequency

<table>
<thead>
<tr>
<th>Digraph</th>
<th>Freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>70</td>
</tr>
<tr>
<td>T</td>
<td>53</td>
</tr>
<tr>
<td>TH</td>
<td>43</td>
</tr>
<tr>
<td>IN</td>
<td>43</td>
</tr>
<tr>
<td>A</td>
<td>42</td>
</tr>
<tr>
<td>T</td>
<td>42</td>
</tr>
<tr>
<td>ER</td>
<td>40</td>
</tr>
<tr>
<td>S</td>
<td>37</td>
</tr>
<tr>
<td>HE</td>
<td>35</td>
</tr>
<tr>
<td>D</td>
<td>34</td>
</tr>
<tr>
<td>G</td>
<td>33</td>
</tr>
<tr>
<td>ON</td>
<td>28</td>
</tr>
<tr>
<td>NG</td>
<td>26</td>
</tr>
<tr>
<td>S</td>
<td>25</td>
</tr>
<tr>
<td>N</td>
<td>25</td>
</tr>
<tr>
<td>AN</td>
<td>24</td>
</tr>
<tr>
<td>GU</td>
<td>24</td>
</tr>
<tr>
<td>H</td>
<td>22</td>
</tr>
<tr>
<td>ST</td>
<td>22</td>
</tr>
<tr>
<td>G</td>
<td>22</td>
</tr>
<tr>
<td>R</td>
<td>22</td>
</tr>
<tr>
<td>N</td>
<td>21</td>
</tr>
<tr>
<td>AT</td>
<td>21</td>
</tr>
<tr>
<td>HA</td>
<td>20</td>
</tr>
<tr>
<td>SE</td>
<td>20</td>
</tr>
<tr>
<td>P</td>
<td>20</td>
</tr>
<tr>
<td>NE</td>
<td>19</td>
</tr>
<tr>
<td>RE</td>
<td>18</td>
</tr>
<tr>
<td>OR</td>
<td>18</td>
</tr>
<tr>
<td>E</td>
<td>17</td>
</tr>
<tr>
<td>M</td>
<td>17</td>
</tr>
<tr>
<td>ES</td>
<td>16</td>
</tr>
<tr>
<td>C</td>
<td>15</td>
</tr>
<tr>
<td>AL</td>
<td>15</td>
</tr>
<tr>
<td>A</td>
<td>15</td>
</tr>
<tr>
<td>EA</td>
<td>14</td>
</tr>
<tr>
<td>ED</td>
<td>14</td>
</tr>
<tr>
<td>ND</td>
<td>14</td>
</tr>
<tr>
<td>OF</td>
<td>14</td>
</tr>
<tr>
<td>CI</td>
<td>14</td>
</tr>
<tr>
<td>TI</td>
<td>14</td>
</tr>
<tr>
<td>EN</td>
<td>14</td>
</tr>
<tr>
<td>TO</td>
<td>14</td>
</tr>
<tr>
<td>AR</td>
<td>14</td>
</tr>
<tr>
<td>IT</td>
<td>14</td>
</tr>
<tr>
<td>F</td>
<td>14</td>
</tr>
<tr>
<td>IC</td>
<td>13</td>
</tr>
<tr>
<td>CO</td>
<td>13</td>
</tr>
<tr>
<td>RO</td>
<td>13</td>
</tr>
<tr>
<td>NT</td>
<td>13</td>
</tr>
<tr>
<td>RT</td>
<td>13</td>
</tr>
<tr>
<td>AD</td>
<td>12</td>
</tr>
<tr>
<td>BE</td>
<td>12</td>
</tr>
<tr>
<td>DE</td>
<td>12</td>
</tr>
<tr>
<td>VE</td>
<td>12</td>
</tr>
<tr>
<td>EX</td>
<td>12</td>
</tr>
<tr>
<td>TE</td>
<td>11</td>
</tr>
<tr>
<td>RA</td>
<td>10</td>
</tr>
<tr>
<td>RC</td>
<td>10</td>
</tr>
<tr>
<td>HI</td>
<td>10</td>
</tr>
<tr>
<td>RS</td>
<td>10</td>
</tr>
<tr>
<td>TA</td>
<td>9</td>
</tr>
<tr>
<td>F</td>
<td>9</td>
</tr>
<tr>
<td>DI</td>
<td>9</td>
</tr>
<tr>
<td>MI</td>
<td>9</td>
</tr>
</tbody>
</table>
## Sample Text 1

<table>
<thead>
<tr>
<th>Rank</th>
<th>Unigraph Freq.</th>
<th>Digraph Freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>T  97</td>
<td>E  22</td>
</tr>
<tr>
<td>2</td>
<td>E  73</td>
<td>T  18</td>
</tr>
<tr>
<td>3</td>
<td>T  47</td>
<td>TH 15</td>
</tr>
<tr>
<td>4</td>
<td>N  41</td>
<td>IN 13</td>
</tr>
<tr>
<td>5</td>
<td>R  36</td>
<td>T  12</td>
</tr>
<tr>
<td>6</td>
<td>L  35</td>
<td>A  11</td>
</tr>
<tr>
<td>7</td>
<td>A  33</td>
<td>HE 11</td>
</tr>
<tr>
<td>8</td>
<td>O  25</td>
<td>Y  11</td>
</tr>
<tr>
<td>9</td>
<td>S  24</td>
<td>YE 9</td>
</tr>
<tr>
<td>10</td>
<td>H  23</td>
<td>J  9</td>
</tr>
<tr>
<td>11</td>
<td>L  18</td>
<td>EN 9</td>
</tr>
<tr>
<td>12</td>
<td>Y  17</td>
<td>S  9</td>
</tr>
<tr>
<td>13</td>
<td>D  17</td>
<td>ED 8</td>
</tr>
<tr>
<td>14</td>
<td>C  13</td>
<td>ER 8</td>
</tr>
<tr>
<td>15</td>
<td>W  10</td>
<td>EA 7</td>
</tr>
<tr>
<td>16</td>
<td>G  10</td>
<td>AR 7</td>
</tr>
<tr>
<td>17</td>
<td>Y  9</td>
<td>S  7</td>
</tr>
<tr>
<td>18</td>
<td>U  9</td>
<td>NT 7</td>
</tr>
<tr>
<td>19</td>
<td>M  8</td>
<td>N  7</td>
</tr>
<tr>
<td>20</td>
<td>F  8</td>
<td>LE 6</td>
</tr>
<tr>
<td>21</td>
<td>D  6</td>
<td>NE 6</td>
</tr>
<tr>
<td>22</td>
<td>B  5</td>
<td>D  6</td>
</tr>
<tr>
<td>23</td>
<td>K  3</td>
<td>R  6</td>
</tr>
<tr>
<td>24</td>
<td>X  1</td>
<td>DE 5</td>
</tr>
<tr>
<td>25</td>
<td>Q  1</td>
<td>NG 5</td>
</tr>
<tr>
<td>26</td>
<td>J  1</td>
<td>H  5</td>
</tr>
<tr>
<td>27</td>
<td>AN 3</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>O  3</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>OR 3</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>A  3</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>G  3</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>B  4</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>BE 4</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>EE 4</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>RE 4</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>SE 4</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>F  4</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>RI 4</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>EL 4</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>N  4</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>TO 4</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>TR 4</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>AS 4</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>RY 4</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Y  4</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>O  4</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>HA 3</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>CC 3</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>ME 3</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>TE 3</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>YE 3</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>OP 3</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>G  3</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>SI 3</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>WI 3</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>AL 3</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>LL 3</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>RL 3</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>M  3</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>NN 3</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>GN 3</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>HO 3</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>RO 3</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>P  3</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>IS 3</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>RS 3</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>AT 3</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>IT 3</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>UT 3</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>IV 3</td>
<td></td>
</tr>
</tbody>
</table>
### Sample Text 2

<table>
<thead>
<tr>
<th>Rank</th>
<th>Unigraph</th>
<th>Freq.</th>
<th>Digraph</th>
<th>Freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E</td>
<td>69</td>
<td>E</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>T</td>
<td>37</td>
<td>T</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>34</td>
<td>T</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>J</td>
<td>25</td>
<td>TH</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>25</td>
<td>A</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>R</td>
<td>24</td>
<td>RE</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>O</td>
<td>23</td>
<td>S</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>N</td>
<td>22</td>
<td>HE</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>H</td>
<td>17</td>
<td>AR</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>U</td>
<td>14</td>
<td>S</td>
<td>7</td>
</tr>
<tr>
<td>11</td>
<td>C</td>
<td>12</td>
<td>IT</td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>M</td>
<td>11</td>
<td>B</td>
<td>6</td>
</tr>
<tr>
<td>13</td>
<td>Y</td>
<td>10</td>
<td>UR</td>
<td>6</td>
</tr>
<tr>
<td>14</td>
<td>Y</td>
<td>9</td>
<td>ST</td>
<td>6</td>
</tr>
<tr>
<td>15</td>
<td>L</td>
<td>8</td>
<td>Y</td>
<td>6</td>
</tr>
<tr>
<td>16</td>
<td>F</td>
<td>8</td>
<td>EA</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>W</td>
<td>7</td>
<td>C</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>B</td>
<td>7</td>
<td>F</td>
<td>5</td>
</tr>
<tr>
<td>19</td>
<td>D</td>
<td>6</td>
<td>I</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>P</td>
<td>5</td>
<td>ON</td>
<td>5</td>
</tr>
<tr>
<td>21</td>
<td>V</td>
<td>3</td>
<td>O</td>
<td>5</td>
</tr>
<tr>
<td>22</td>
<td>G</td>
<td>3</td>
<td>AT</td>
<td>5</td>
</tr>
<tr>
<td>23</td>
<td>Q</td>
<td>1</td>
<td>TI</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>O</td>
<td>1</td>
<td>OM</td>
<td>4</td>
</tr>
<tr>
<td>25</td>
<td>C</td>
<td>1</td>
<td>CO</td>
<td>4</td>
</tr>
<tr>
<td>26</td>
<td>E</td>
<td>1</td>
<td>ES</td>
<td>4</td>
</tr>
<tr>
<td>27</td>
<td>I</td>
<td>1</td>
<td>IC</td>
<td>3</td>
</tr>
<tr>
<td>28</td>
<td>N</td>
<td>1</td>
<td>ND</td>
<td>3</td>
</tr>
<tr>
<td>29</td>
<td>M</td>
<td>1</td>
<td>ME</td>
<td>3</td>
</tr>
<tr>
<td>30</td>
<td>E</td>
<td>1</td>
<td>NE</td>
<td>3</td>
</tr>
<tr>
<td>31</td>
<td>S</td>
<td>1</td>
<td>SE</td>
<td>3</td>
</tr>
<tr>
<td>32</td>
<td>V</td>
<td>1</td>
<td>VE</td>
<td>3</td>
</tr>
<tr>
<td>33</td>
<td>I</td>
<td>1</td>
<td>IL</td>
<td>3</td>
</tr>
<tr>
<td>34</td>
<td>A</td>
<td>1</td>
<td>AN</td>
<td>3</td>
</tr>
<tr>
<td>35</td>
<td>N</td>
<td>1</td>
<td>EN</td>
<td>3</td>
</tr>
<tr>
<td>36</td>
<td>I</td>
<td>1</td>
<td>IN</td>
<td>3</td>
</tr>
<tr>
<td>37</td>
<td>O</td>
<td>1</td>
<td>IO</td>
<td>3</td>
</tr>
<tr>
<td>38</td>
<td>T</td>
<td>1</td>
<td>TR</td>
<td>3</td>
</tr>
<tr>
<td>39</td>
<td>O</td>
<td>1</td>
<td>OS</td>
<td>3</td>
</tr>
<tr>
<td>40</td>
<td>T</td>
<td>1</td>
<td>NT</td>
<td>3</td>
</tr>
<tr>
<td>41</td>
<td>W</td>
<td>1</td>
<td>W</td>
<td>3</td>
</tr>
<tr>
<td>42</td>
<td>A</td>
<td>1</td>
<td>A</td>
<td>3</td>
</tr>
<tr>
<td>43</td>
<td>D</td>
<td>1</td>
<td>D</td>
<td>3</td>
</tr>
<tr>
<td>44</td>
<td>R</td>
<td>1</td>
<td>R</td>
<td>3</td>
</tr>
<tr>
<td>45</td>
<td>N</td>
<td>1</td>
<td>NA</td>
<td>2</td>
</tr>
<tr>
<td>46</td>
<td>E</td>
<td>1</td>
<td>EC</td>
<td>2</td>
</tr>
<tr>
<td>47</td>
<td>D</td>
<td>1</td>
<td>DE</td>
<td>2</td>
</tr>
<tr>
<td>48</td>
<td>Y</td>
<td>1</td>
<td>YE</td>
<td>2</td>
</tr>
<tr>
<td>49</td>
<td>G</td>
<td>1</td>
<td>GF</td>
<td>2</td>
</tr>
<tr>
<td>50</td>
<td>C</td>
<td>1</td>
<td>CH</td>
<td>2</td>
</tr>
<tr>
<td>51</td>
<td>H</td>
<td>1</td>
<td>HI</td>
<td>2</td>
</tr>
<tr>
<td>52</td>
<td>I</td>
<td>1</td>
<td>RI</td>
<td>2</td>
</tr>
<tr>
<td>53</td>
<td>M</td>
<td>1</td>
<td>SI</td>
<td>2</td>
</tr>
<tr>
<td>54</td>
<td>O</td>
<td>1</td>
<td>MO</td>
<td>2</td>
</tr>
<tr>
<td>55</td>
<td>R</td>
<td>1</td>
<td>RO</td>
<td>2</td>
</tr>
<tr>
<td>56</td>
<td>S</td>
<td>1</td>
<td>IS</td>
<td>2</td>
</tr>
<tr>
<td>57</td>
<td>S</td>
<td>1</td>
<td>SS</td>
<td>2</td>
</tr>
<tr>
<td>58</td>
<td>U</td>
<td>1</td>
<td>US</td>
<td>2</td>
</tr>
<tr>
<td>59</td>
<td>T</td>
<td>1</td>
<td>UT</td>
<td>2</td>
</tr>
<tr>
<td>60</td>
<td>C</td>
<td>1</td>
<td>CU</td>
<td>2</td>
</tr>
<tr>
<td>61</td>
<td>O</td>
<td>1</td>
<td>OU</td>
<td>2</td>
</tr>
<tr>
<td>62</td>
<td>S</td>
<td>1</td>
<td>SU</td>
<td>2</td>
</tr>
<tr>
<td>63</td>
<td>R</td>
<td>1</td>
<td>RV</td>
<td>2</td>
</tr>
<tr>
<td>64</td>
<td>W</td>
<td>1</td>
<td>OW</td>
<td>2</td>
</tr>
<tr>
<td>65</td>
<td>E</td>
<td>1</td>
<td>EY</td>
<td>2</td>
</tr>
<tr>
<td>66</td>
<td>Y</td>
<td>1</td>
<td>EY</td>
<td>2</td>
</tr>
<tr>
<td>67</td>
<td>N</td>
<td>1</td>
<td>N</td>
<td>2</td>
</tr>
<tr>
<td>68</td>
<td>T</td>
<td>1</td>
<td>TY</td>
<td>2</td>
</tr>
<tr>
<td>69</td>
<td>Y</td>
<td>1</td>
<td>Y</td>
<td>2</td>
</tr>
</tbody>
</table>
APPENDIX E

*IMS Demonstrator module descriptions and source listings*
Appendix E: IMS Demonstrator module descriptions and source listings

The IMS Demonstrator system was developed using *Borland Turbo C for DOS* (version 3.0), running on an IBM PC-compatible system.

A summary of the various code modules involved is presented in the table overleaf. In each case the title of the module is given, along with an indication of its type (normally an executable program or a function / code library) and a brief description. The majority of the descriptions are taken from directly from the module headers.

The full source code for each of these modules is provided on the accompanying diskette that is bound with the thesis.
<table>
<thead>
<tr>
<th>Module</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MENU</td>
<td>Executable</td>
<td>The IMS menu system. Specifies each of the user menus and also includes code for the various checklists and configuration options found in the system. Creates default IMS configuration file if one does not already exist. All other modules associated with the Host side of the demonstrators functionality are called from this module.</td>
</tr>
<tr>
<td>CLIENT</td>
<td>Executable</td>
<td>Keystroke monitoring and authentication module for IMS. Runs on local PC as Terminate Stay Resident (TSR) code. Keystroke timings transmitted to remote SMC for analysis. Collects current system configuration data, performs virus scan and accepts user id and password (validated by IMS Host) before entering TSR state. Sends regular signal to Host whilst operational to it to detect when Client system is reset.</td>
</tr>
<tr>
<td>HOST</td>
<td>Executable</td>
<td>IMS Host monitoring / authentication module. Monitors keystrokes from IMS client and compares to profile. Profile comparison performed dynamically in real-time. Plots real-time graph of test subject performance. Client system configuration audit, virus scan and user authentication performed before monitoring session begins. Can detect reset of Client system and restart monitoring.</td>
</tr>
<tr>
<td>MONI_SET</td>
<td>Executable</td>
<td>Allows update of monitor configuration settings (settings used by HOST).</td>
</tr>
<tr>
<td>PROF_SET</td>
<td>Executable</td>
<td>Allows update of profiler configuration settings (settings used by REG USER).</td>
</tr>
<tr>
<td>REG_USER</td>
<td>Executable</td>
<td>User profiling / registration module. Calculates &amp; stores keystroke characteristics of named users. Also stores password and question / answer for use in logins and authentication challenges. Filters out high &amp; low 10% of sample times for digraphs if standard deviation of times exceeds the mean. Reports profiling statistics and typist classification to a file. Automatically detects end of sample input &amp; disables RETURN.</td>
</tr>
<tr>
<td>VIEWUSER</td>
<td>Executable</td>
<td>Allows inspection of the IMS profile database.</td>
</tr>
<tr>
<td>TESTPROF</td>
<td>Executable</td>
<td>Obtains test samples from profiled users in order to determine user-specific authentication thresholds. Profiles are then updated with the new values.</td>
</tr>
<tr>
<td>UPD_USER</td>
<td>Executable</td>
<td>Allows update of user record in IMS profile database. Can modify password, challenge data and authentication thresholds.</td>
</tr>
<tr>
<td>DEL_USER</td>
<td>Executable</td>
<td>Allows deletion of user record from IMS profile database.</td>
</tr>
<tr>
<td>Module (cont.)</td>
<td>Type (cont.)</td>
<td>Description (cont.)</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>SET_TEXT</td>
<td>Executable</td>
<td>Accept text pattern for use in user profiling or test sampling. Statistical analysis of unigraph and digraph frequencies stored in a report file.</td>
</tr>
<tr>
<td>TEXTEDIT</td>
<td>Executable</td>
<td>Allows an existing IMS text to be edited (using DOS EDIT). This option is intended to allow a means of correcting typing errors etc from the original text entry as opposed to a tool for making major changes.</td>
</tr>
<tr>
<td>TEXTANAL</td>
<td>Executable</td>
<td>Standalone text analyser to allow existing text files to be assessed for suitability for IMS usage. Statistical analysis of unigraph and digraph frequencies stored in a report file.</td>
</tr>
<tr>
<td>VIEW_LOG</td>
<td>Executable</td>
<td>Allows inspection of the IMS audit log. Includes limited search facility based on date field.</td>
</tr>
<tr>
<td>IMS_HEAD</td>
<td>Header</td>
<td>Common definitions required by other IMS modules. This is included at the start of all of the source code for all executable modules.</td>
</tr>
<tr>
<td>IMS_FUNC</td>
<td>Library</td>
<td>General functions called by other IMS modules. Whilst originally developed for use in IMS, these functions might also be usefully employed in other applications. Examples include routines for direct video output, character code conversion and menu display.</td>
</tr>
<tr>
<td>IMSINCL2</td>
<td>Library</td>
<td>Common routines for inclusion in other modules (conceptually distinct from IMS_FUNC in that the routines are all IMS-specific).</td>
</tr>
<tr>
<td>IMS_GRPH</td>
<td>Library</td>
<td>Graph routines used by the IMS demonstrator. These are also coded in an IMS-specific manner, and are separated out primarily to improve the readability of the HOST module.</td>
</tr>
</tbody>
</table>
During the course of this research the author has contributed to eight published papers, as detailed below.


Appendix F: List of Publications


In addition, two further papers have been written relating to the security guidelines for existing healthcare systems that were developed for the SEISMED project and described in chapter 4:


Copies of each of these papers are bound within this appendix of the thesis.
Data security in medical information systems using a generic model

P. Sanders and S. Furnell

University of Plymouth, Plymouth, U.K.

Abstract

The content of this paper is based upon work currently being carried out as part of the Commission of European Communities SEISMED (Secure Environment for Information Systems in MEDicine) project, the aim of which is to provide recommendations on security for existing systems in European Health Care Establishments (HCEs).

1. Introduction

The need for adequate data security in the medical environment is obvious, given that the maintenance of patient confidentiality and safety are of paramount importance to retaining a relationship of trust between patients and the HCE. In addition, the transition to the purchaser-provider system of funding now present in parts of the European Community means that more traditional business-type data also require protection.

A number of methods of protection may be suitable for adoption in the medical field, ranging from technical measures (achieved either via software or hardware) on the systems themselves to procedures implemented across the HCE [1]. In broad terms the methods fall into 3 main categories, as below:

- **External control mechanisms**
  Safeguards against fire, flood, theft, equipment or power failure and such like.
  Emphasis of security through staff awareness programmes.

- **User Interface control mechanisms**
  Provision of authentication / access control features (e.g. the use of passwords, tokens, and related issues).

- **Internal control mechanisms**
  Including such concepts as data encryption, virus prevention, system auditing.

These general ideas have been explored in detail in previous publications in a piecemeal approach. What is now required is a set of guidelines on where, what and how to put security into HCE systems in general. It would then be possible for individual system administrators to select solutions appropriate for their own particular arrangements.

The provision of security for medical data on a large scale is a complex issue, given that a myriad of different computer systems (in terms of hardware, networking and actual applications) may be identified within a single country, let alone in the full European scenario. The issue is further complicated by the variety of information that may be held, and the fact that several different levels of sensitivity may exist. As the desired protection will depend upon the risks associated with the information, it is impossible to assert a single level of security that will be appropriate for all data.

In order to address these problems there is a requirement for a flexible system which is able to integrate security into the multiple networks and databases in an open systems type environment. In addition, a method is needed to simplify the identification of security requirements for individual systems.

2. Method of Implementation

In consultation with a number of Health Care Establishments (HCEs) within Europe, the general care activities carried out by hospitals, general practitioners, community health care centres, and various other support services have been examined. This has enabled a generic model of the medical
environment to be produced that can be used as the basis for further investigation [2].

The analysis established that, at a high level, all medical environments are of a similar nature (i.e. their aim is to provide a very similar set of services, albeit in slightly different ways, with differing levels of sophistication). The activities involved in the provision of health care can be seen to fall into the basic sequence of operations shown in Fig. 1.

At each stage of this sequence a variety of patient care or administration data may be generated or utilised from existing knowledge (i.e. medical or organisational). The type and quantity of information involved will be dependant upon the problems and requirements of the individual patients. In addition, the support services that surround the main care activities may also produce or use further data of their own.

This information may be of varying levels of sensitivity, and this will again be highly dependent upon the cases involved. Data relating to the clinical side of care delivery may be considered to fall into four main classifications in terms of sensitivity:

**Operational:**
Information used directly to make / govern patient care decisions. Can be subdivided into:
- General (the vast majority of patients)
- Special (e.g. HCE staff or special groups in the community)
- Sensitive (e.g. patients with sensitive problems such as AIDS or psychological disorders).

**Non-Operational:**
Information that does not directly govern patient decisions but is used for planning and resourcing purposes (e.g. analysis of workloads).

An overall view of the data involved is given by grouping them into the categories shown in Fig. 2. Obviously the categories shown are of a (necessarily) broad nature, but they may be broken down into further levels of detail as required. For example:

**Patient Care**
This group would contain the medical history, diagnosis, care decisions and treatment information that relate to individual patients. Data examples could be:
- Episode information: Specific needs
- Dates of admissions / discharges: Health care delivered
- Staff involved: Drug therapy
- Diagnosis including clinical coding/s: Outcome of the treatment
- Care plan: Consultant and anesthetist reports

The above groups now provide a generic framework encompassing all data required by a HCE. Specific medical applications may utilise information from all of the data groups, or simply a subset of them. It is consequently possible to map such applications onto the model, indicating the data groups that are involved. This can be used to highlight any weaknesses in the systems, and hence suggest the security services that may be required.

To illustrate how this mapping may be achieved,
Fig. 3 shows how the Patient Administration System (PAS), as used by the Plymouth Health Authority, can be incorporated into such an arrangement.

At this stage the risks or threats that may be associated with each type of data in the system may be considered in terms of the core elements of security: disclosure of the information to either HCE staff or outsiders (confidentiality), denial of access to the information over various periods (availability) and modification or destruction of data (integrity) and user authentication. Several categories of risk can be identified, all of which must be considered in order to determine how serious their impact would be in each case:

- Commercial confidentiality
- Disruption
- Embarrassment
- Financial Loss
- Legal
- Personal privacy
- Safety

For example, the disclosure of sensitive patient care information to HCE outsiders could be seen as a serious risk in terms of legal action, patient personal privacy and embarrassment to both the patient and the HCE.

Each category of risk suggests certain protection
measures that should be incorporated into a system. For example, in the cases of Embarrassment and Safety the following security services are suggested:

**Embarrassment:**
This requires a low to medium confidentiality service to be provided. In a low level system, standard password authentication with access limitation may be appropriate. For medium confidentiality the addition of card identification and audit may be more practical.

**Safety:**
This is the most important aspect from the patient care viewpoint, and warrants the highest possible levels of integrity as well as a strong backup source. Use of check codes and encryption, as well as full auditing and a high level of user authentication seems necessary.

A practical method of realising these security services in existing or new systems is to incorporate them as an add-on service. The use of a
 Comprehensive Integrated Security System (CISS) overlay arrangement implemented in a modular fashion would allow the provision of a full range of services/mechanisms to users and applications [3]. The security requirements of a range of systems could be catered for by appropriate combinations of the generalised modules, thus allowing sufficient flexibility to suit different computing environments. The development of the CISS on a standardised architecture, such as the ISO OSI (Open Systems Interconnection) model, would in turn facilitate integration between different security domains.

3. References

A generic methodology for health care data security

S. M. FURNELL†, P. N. GAUNT‡, G. PANGALOS§, P. W. SANDERS† and M. J. WARREN†
† Security Research Group, Faculty of Technology, University of Plymouth, UK
‡ Department of Health Care Informatics, University of Plymouth/Derriford Hospital, UK
§ Informatics Laboratory, Faculty of Technology, Aristotelian University of Thessaloniki, Greece

(Received November 1993)

Abstract. The aim is to outline the framework of a generic methodology for specifying countermeasures in health care environments. The method is specifically aimed at the enhancement of security in existing health care systems, and a key element is the use of predetermined 'profiles' by which these may be classified. Example scenarios are presented to illustrate how the concept could be applied in practice. The paper is based upon work that was initially carried out as part of the Commission of European Communities SEISMED (Secure Environment for Information Systems in MEDicine) project, the aim of which is to provide security recommendations for European health care establishments (HCEs).

Keywords: Risk analysis; System profiling.

1. Introduction

During the past few decades the use of information technology (IT) has become more widespread in all areas of society, and the types of activities that it performs or supports have become increasingly more important. As a result, information systems are now heavily utilized by all levels of staff, and relied upon to the extent that it would be difficult to manage without them.

The health care field has been no exception to the trend, as witnessed by the wide variety of applications that now handle many types of health data [1]. These systems contain vast amounts of information, much of it relating to individuals and of a sensitive nature. In addition to direct care applications, some parts of the European Community are now making the transition to a purchaser–provider funding system, meaning that an increasing volume of traditional business type data must also be maintained.

The combination of these points serves to make the protection of health information systems a vital concern, and necessitates that security is now considered as an essential aspect of the information technology field.

At a high level, information security is defined as being the combination of the following key factors [2]:

(1) Confidentiality. This refers to the prevention of unauthorized disclosure of information. All access to data must be restricted to authorized users who have a legitimate 'need to know'. Confidentiality is fundamental in health care since certain categories of data may be of a particularly sensitive nature,
and disclosure could result in significant embarrassment or prejudice to the individual concerned.

(2) Integrity. The prevention of unauthorized modification of information. There is a requirement to be able to trust the system and be confident that the same information can be retrieved as was originally entered. For example, the accidental or deliberate alteration of patient-related data could have serious implications for care delivery.

(3) Availability. Data and systems should be accessible and usable (by authorized users) when and where they are required. This requirement necessitates both prevention of the unauthorized withholding of information or resources, and adequate safeguards against system failure. In some medical environments, for example, critical systems may be required to be in operation 24 h a day, 7 days a week.

Security breaches may result from a variety of accidental or deliberate acts, with potential threats being posed by outsiders and from staff within the organization. Deliberate acts may include activities such as fraud, theft, hacking and virus infection. The health care field has certainly not been immune to these threats, with the most recent UK survey [3] showing that 10% of reported security incidents were related to health care systems (with roughly an even split between the above categories).

The introduction of information security seeks to eliminate or, more realistically, reduce the vulnerability to any risks that may be present. Protection must encompass the computer system and everything associated with it (e.g. from the computer unit itself to the building in which it is housed). Most important, however, is the protection of the information stored in the systems. These goals may be realized via a variety of measures [4], of both a technical and non-technical nature (e.g. physical, personnel and administrative controls).

In a health care establishment (HCE), any part of the computing system could provide the basis for a security breach, and this multiplicity of targets makes medical security a difficult issue. Large-scale introduction is complicated by the myriad of different system configurations (in terms of hardware, networking and actual applications) that may be identified within a single country, let alone within the full European scenario [5]. The issue is further complicated by the variety of information that may be held, and the fact that several different levels of data sensitivity may exist. The desired protection will depend upon several factors including the computer configuration, the operational environment and the information itself. As such it is impossible to assert a single level of security that will be appropriate for all cases without it being excessive in some applications.

Introducing security is a balancing process between providing the desirable level of protection against the maintenance of an adequate level of availability and performance (so that legitimate users have easy access to the data). Specifying the level of security that should be included involves some judgement about the dangers associated with the system, the required level of availability and the resource implications of various means of avoiding or minimizing those dangers.

Guidelines are therefore required on the selection of appropriate security measures, as well as on where and how to put them into HCE systems in general. The commonly accepted means of achieving this is to conduct a risk analysis investigation. However, this can be a time-consuming and costly proposition, and
may consequently be prohibitive in many cases. It would obviously be undesirable for security to be overlooked when this occurs. Given that many of the threats and vulnerabilities of individual HCEs are not unique, a full risk analysis in each case may also be largely unnecessary.

This paper proposes the framework of a methodology that is able to simplify the identification of security requirements for individual systems. This provides a straightforward means by which system administrators/security officers can select solutions appropriate for their own particular arrangements.

2. A conceptual overview of the generic methodology

Security should be examined from the perspective of the whole system, with all factors that influence protection requirements being considered. In general terms the security-relevant elements of existing systems are characterized as follows:

Information system = Computer configuration + Operational environment + Data sensitivity

These elements have been incorporated into the framework of a system protection methodology as shown in figure 1. This illustrates (at a high level) the steps involved in profiling existing systems to determine their requirements and select appropriate countermeasures.

The rationale of the methodology is that similar organizations/systems will have similar security requirements and a key factor in the approach was to devise a number of predetermined security 'profiles' for each element of existing systems. What the methodology proposes is a 'mix-and-match' approach to countermeasure selection, based upon a comparison of existing systems against general profiles. Using appropriate combinations it is possible, at a high level, to generate existing system profiles/categorizations that could then account for the majority of health care IT scenarios. From these it should be feasible to specify appropriate protection measures to meet the security requirements in each case.

The main elements of the methodology are now considered in more detail.

2.1. Computer configuration

This refers to the IT assets (both hardware and software) of the organization. At a high level it is possible to identify a relatively small number of elements which may be included in any given computer configuration, as shown in figure 2. Individual systems would be considered to determine which elements are applicable, and countermeasures selected accordingly. Examples of associated baseline countermeasures have been identified for each configuration, and are grouped as shown in table 1.

2.2. Operational environment

This considers the nature of the environment in which the IT assets are actually located and used, which may also affect the type and level of protection that is required. Table 2 indicates the main environmental considerations that may have security bearing. Appropriate combinations of these factors can be used to describe the majority of health care establishments (i.e. from GPs to general hospitals). Again, appropriate baseline countermeasures can be specified for each type of environment, and the key issues are indicated in table 3.
2.3. Data sensitivity

The sensitivity of data is determined by two major factors, as shown in figure 3. These factors, and the means of rating sensitivity, will now be considered in more detail.

2.3.1. Data type. In consultation with a number of HCEs within Europe, the general care activities carried out by hospitals, general practitioners, community health care centres, and various other support services were examined. This enabled a generic model of medical data to be developed as the basis for further investigation [6]. The model is composed of 12 main data groups, as described in table 4. The purpose is to allow a simple means of specifying what data are available within
Table 2. Operational environment categories.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Options</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Fixed/mobile</td>
<td>Variable environment (e.g. portable computer system) limits environmental measures</td>
</tr>
<tr>
<td>Rural/urban/city</td>
<td></td>
<td>Local environment is an indicator of local population density, crime potential and likelihood of natural disasters</td>
</tr>
<tr>
<td>Buildings</td>
<td>Single/multiple</td>
<td>Number of buildings will determine access control, site security requirements</td>
</tr>
<tr>
<td>Old/modern</td>
<td></td>
<td>Age of building may indicate risk of fire, natural damage, etc.</td>
</tr>
<tr>
<td>People</td>
<td>Number</td>
<td>Number and mixture of people influences access controls and personnel-related measures</td>
</tr>
<tr>
<td></td>
<td>(low, medium, high)</td>
<td></td>
</tr>
<tr>
<td>Staff/contract/public</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Operational environment countermeasure categories.

<table>
<thead>
<tr>
<th>Category</th>
<th>Example issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site security</td>
<td>Building/site access, theft prevention</td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Fire, flood, natural disasters</td>
</tr>
<tr>
<td>Procedural</td>
<td>Control of visitors, controls on smoking, eating/drinking</td>
</tr>
<tr>
<td>Personnel</td>
<td>Job recruitment/termination, awareness</td>
</tr>
</tbody>
</table>

Figure 3. Factors of data sensitivity.

a system and help in the allocation of appropriate sensitivities, thus simplifying the process of identifying how and where data are located in different computer systems and networks. The information used by the HCE may be of varying levels of sensitivity, and this will again be highly dependent upon the cases involved.

The models groups are of a (necessarily) broad nature, but they may be broken down into further levels of detail as required. For example:

*Patient care*: Episode information, Dates of admissions/discharges, Staff involved, Diagnosis including clinical codings(s), Care plan, Specific needs, Health care delivered, Drug therapy, Outcome of the treatment, Consultants' and anaesthetists' reports.

The model provides a generic framework that should encompass all data required by a HCE. Specific medical applications may store and communicate information from all of the data groups, or a particular subset of them. It is consequently possible to map such applications on to the model, indicating the data groups that are
Table 4. Generic data group descriptions.

<table>
<thead>
<tr>
<th>Data group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient identification</td>
<td>General information held regarding individual patients referred to the health care service. Often utilized by a number of different systems/applications</td>
</tr>
<tr>
<td>Patient administration</td>
<td>Information used in the day-to-day scheduling of various non-clinical care activities related to patients (i.e. concerned with the delivery of resources that in turn facilitate clinical care)</td>
</tr>
<tr>
<td>Patient care</td>
<td>Contains medical history, diagnosis, care decisions and treatment information relating to individual patients</td>
</tr>
<tr>
<td>Clinical services</td>
<td>Information related to the functioning of service departments of the HCE. Data are for the department’s internal use (not patient-related)</td>
</tr>
<tr>
<td>Finance</td>
<td>Information covering all aspects of finance that are involved in the operation of HCEs</td>
</tr>
<tr>
<td>Hotel services</td>
<td>Information stored on all the basic ‘housekeeping’ functions of health care systems</td>
</tr>
<tr>
<td>Staff</td>
<td>Personnel information relating to all grades of HCE staff</td>
</tr>
<tr>
<td>Resource management and planning</td>
<td>Information used in the management, monitoring and planning of health care organizations</td>
</tr>
<tr>
<td>Library and information services</td>
<td>Encompasses the existing medical knowledge that is referenced by clinical staff, and national/local protocols for clinical management</td>
</tr>
<tr>
<td>Expert systems</td>
<td>Information utilized by decision support tools and/or neural networks within the HCE</td>
</tr>
<tr>
<td>Communication services</td>
<td>Identifies the process of communication within the HCE. Could contain a variety of additional data generated during organizational communication (e.g. activity requests, transaction information)</td>
</tr>
<tr>
<td>External systems</td>
<td>Recognizes potential data relationships (interfaces) that may exist with other HCE applications/systems</td>
</tr>
</tbody>
</table>

Table 5. General categories of medical data usage.

<table>
<thead>
<tr>
<th>Data use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational clinical</td>
<td>Planning, delivery and monitoring of health care</td>
</tr>
<tr>
<td>Emergency care</td>
<td>Provision of care in a clinical emergency, where optimal conditions/information cannot be guaranteed</td>
</tr>
<tr>
<td>Critical clinical</td>
<td>Control of instrumentation/systems in direct feedback loops</td>
</tr>
<tr>
<td>Expert systems</td>
<td>Use in decision support tools or neural networks</td>
</tr>
<tr>
<td>Operational non-clinical</td>
<td>Supporting HCE infrastructure, but not directly influencing care of individuals</td>
</tr>
<tr>
<td>Financial</td>
<td>Contract management, purchasing and billing</td>
</tr>
<tr>
<td>Planning and resource management</td>
<td>Aggregation of data for planning and review purposes</td>
</tr>
<tr>
<td>Quality management</td>
<td>Clinical audit, assessment of care efficiency and outcome</td>
</tr>
<tr>
<td>Clinical research</td>
<td>Identifiable or anonymized data used for research purposes: usually utilizes aggregated data</td>
</tr>
</tbody>
</table>
involved, and from this derive the basic sensitivity of the information. Examples of such mappings are given later in the text.

2.3.2. Data use. Incorporating this factor of data sensitivity into the methodology demands that an appropriate range of general uses can be identified. Related work within the SEISMED project [7] has determined a high-level set of data uses that are appropriate for our purposes. A total of nine categories is considered, as described in table 5.

2.3.3. Sensitivity ratings. Sensitivity is quantified in terms of several different types of impact that may relate to the data in the system. Four main types of impact can be identified, with appropriate countermeasures being given in each case.

1. Disclosure. Unauthorized disclosure of information to HCE staff or outsiders.
2. Denial. Denial of access to the information for varying periods.
3. Modification. Accidental or deliberate alteration of the information.

The type and use of the data will have different influences over the protection requirements in each of these cases.

Disclosure. Data type is the most significant factor in determining the confidentiality requirement, as data will generally portray the same information in all contexts. The protection afforded should therefore remain constant regardless of which application uses it. However, data usage may still have some effect as it can influence problems arising through data aggregation. It is conceivable that, if certain data elements are combined, then the impact of disclosure may be greater than that of any one element in isolation.

Denial, modification and destruction. The requirements for these are primarily determined by the data usage, as the context will determine the seriousness of the impact.

Impacts are rated low, medium or high (where low indicates that the baseline countermeasure level is satisfactory, and high is the maximum protection that can be provided). The level is determined by considering a number of potential influencing factors: (a) confidentiality (both personal and commercial), (b) disruption, (c) embarrassment, (d) financial loss, (e) legal, (f) personal safety. For example, the disclosure of sensitive patient care information to HCE outsiders could be seen as a serious risk in terms of legal action, patient personal privacy and embarrassment to both the patient and the HCE. The level of impact will in turn determine the level of countermeasure.

Medical opinion from within various European HCEs was sought in obtaining the impact valuations (using a small survey distributed to appropriate personnel). Nevertheless, it is recognized that, because of the inherent subjectivity in any judgements (based largely on individual roles and/or perceptions of the problems), the resulting figures represent 'reasonable' rather than 'correct' values (i.e. values which the majority of health care professionals would be prepared to accept as an adequate representation of the situation).
2.4. Other factors

This element of the methodology highlights the fact that whilst the 'appropriate countermeasures' suggested may be suitable when considering the existing system in isolation, a number of real-world factors are also likely to influence the final selection process. Such factors are principally considered to include the following:

1. **Cost constraints.** The cost of adopting particular countermeasures may be considered from several angles (e.g. financial, performance, practicality, etc.). The acceptable levels will obviously be highly dependent upon individual environments and their priorities. Financial cost is perceived as being a particularly key factor in security-related decision-making for the majority of health care establishments.

2. **Operational constraints.** The selection of countermeasures will also be influenced by the nature of the organization itself. Any proposals must fit in with what is likely to be tolerated/accepted within the particular health care environment, and should not conflict too greatly with established practice. This relates to the 'business culture' of the organization.

3. **Existing countermeasures.** Any security countermeasures that are already in place in relation to the existing system will obviously influence whether some of the suggested countermeasures need to be considered/adopted.

These would obviously be very subjective elements in the application of the methodology, and it is not possible to formalize them further.

2.5. Countermeasures

Actual security countermeasures are identified and refined at various stages within the methodology, and it can be seen from figure 1 that they are categorized under three headings. These are distinguished as shown below:

1. **Baseline countermeasures.** Represents the minimal security considerations for a given computer configuration in a particular environment, and should be considered irrespective of the data held or the purpose(s) the system is used for.

2. **Appropriate countermeasures.** Represents the overall set of countermeasures that may be appropriate for a given system, considering what data are used and how, but not taking into account any practical constraints that may apply in respect to implementation.

3. **Selected countermeasures.** Represents the final output of the methodology, namely a set of countermeasures that may be added to the existing system to address the security requirements (having considered any limitations of the individual HCE).

The countermeasures used with the methodology are derived from a representative set that are being developed for use within the SEISMED project [8].

3. Methodology implementation

This section describes the specific steps by which the methodology would be implemented when considering individual existing systems.

In order to apply the method the following factors would need to be identified for the specific system/application being considered: (a) computer configuration involved, (b) type of operational environment(s), (c) data groups involved,
(d) purpose of application (data use(s)). Countermeasures would then be derived as shown in figure 4. At each stage appropriate countermeasures would be selected from corresponding categories (NB: It is likely that some duplication may occur in terms of the countermeasures suggested within different categories).

The stages of the methodology may be more formally described as follows:

**Stage 1: Determine basic system profile**
*Input*: none.
*Output*: baseline countermeasures.
*Description*: categorize computer configuration and operational environment of the existing system according to predetermined profile categories. For computer configuration choose appropriate elements from: (a) laptop/portable, (b) desktop PC, (c) mini/mainframe, (d) network. For operational environment categorize elements of: (a) location, (b) buildings, (c) people.

**Stage 2: Determine data sensitivity**
*Input*: none.
*Output*: data-related countermeasures.
*Description*: establish data types and uses. Select countermeasures based upon sensitivities encompassed. Choose appropriate levels from each of: (a) disclosure countermeasures, (b) denial countermeasures, (c) modification countermeasures, (d) destruction countermeasures. This stage is described in more detail below.

**Stage 3: Determine appropriate system countermeasures**
*Input*: baseline countermeasures, data-related countermeasures.
*Output*: appropriate system countermeasures.
*Description*: generate countermeasure set that would satisfy the requirements of the existing system.

**Stage 4: Select system countermeasures**
*Input*: appropriate countermeasures.
*Output*: selected (final) system countermeasures.
*Description*: refine countermeasure set by considering any HCE specific factors/constraints that may apply.
3.1. Determining data sensitivity

Determining the data sensitivity countermeasures for an existing system is the most complex stage of the methodology, as they will be based upon a variety of impact values derived from the data involved. All data groups in the system must be considered to establish: (a) impact valuations for disclosure (based on data type only); (b) impact valuations for denial, modification, destruction (based on data type and use). The specific procedure involved is illustrated in figure 5. These stages and descriptions are listed below:

2.1. Identify the data groups involved using generic data model.
2.2. Determine disclosure impacts from model group valuations.
2.3. Identify general data usage category(s) that applies to the system.
2.4. Determine denial, modification and destruction impacts from usage valuations for each data group involved.
2.5. Derive overall sensitivity values for application by selecting 'worst-case' values from component groups (four values in total).
2.6. Determine appropriate data sensitivity countermeasures using values from 2.5.

4. Illustrative examples

The following section presents two basic examples to illustrate how the
methodology may be applied in practice. These are based on typical information system scenarios that may be found within the UK health service.

Note that the countermeasures and impact levels given in the examples are selected from predetermined lists. However, listing a full set of countermeasures is outside the scope of this paper, and the examples therefore provide only a small representative selection. It should also be noted that the examples only proceed to stage 3 of the methodology. The reason for this is that stage 4 is very much related to the subjective factors of real-world environments, and imposing artificial constraints would add little to the examples.

4.1. Example 1

4.1.1. Scenario. A patient records system maintained by a small GP practice. The system is primarily based upon a standalone PC, although selected data may be transferred to and from this using a portable computer that the GP takes on general visits and emergency call-outs. The practice is based in a single, modern building located in an inner city.

4.1.2. Methodology implementation

Stage 1: Determine basic system profile
Computer configuration: Laptop/portable—standalone; Desktop PC—standalone.
Operational environment: Location—fixed and mobile, city; building—single, modern; People—staff, public, low.

Stage 2: Determine data sensitivity

Stage 2.1: Identify data groups. Three data groups are encompassed, and can be identified from the existing model as shown in figure 6.

Stage 2.2: Determine disclosure impacts

<table>
<thead>
<tr>
<th>Data group</th>
<th>Impact level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient identification</td>
<td>Low</td>
</tr>
<tr>
<td>Patient administration</td>
<td>Medium</td>
</tr>
<tr>
<td>Patient care</td>
<td>High</td>
</tr>
</tbody>
</table>

Stage 2.3: Identify data uses. Potential data uses are identified as follows: (a) operational clinical, (b) emergency care.

Stage 2.4: Determine denial, modification and destruction impacts

<table>
<thead>
<tr>
<th>Data group</th>
<th>Use</th>
<th>Denial</th>
<th>Modification</th>
<th>Destruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient identification</td>
<td>Operational clinical</td>
<td>Medium</td>
<td>Medium</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Emergency care</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
</tr>
<tr>
<td>Patient administration</td>
<td>Operational clinical</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Emergency care</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Patient care</td>
<td>Operational clinical</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>Emergency care</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
</tr>
</tbody>
</table>

Stage 2.5: Derive overall sensitivity ratings. The 'worst-case' impacts from the previous tables are extracted to determine the overall sensitivity: disclosure, high; denial, medium; modification, high; destruction, high.

Stage 3: Determine appropriate system countermeasures

Computer configuration

<table>
<thead>
<tr>
<th>Countermeasure category</th>
<th>Laptop/portable (standalone)</th>
<th>Desktop PC (standalone)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>Casing locks</td>
<td>Locks and/or alarms</td>
</tr>
<tr>
<td></td>
<td>Property markings (visible and UV)</td>
<td>Property markings (visible and UV)</td>
</tr>
<tr>
<td></td>
<td>Protective carry case</td>
<td>Site to deny casual access</td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Service warranty</td>
<td>On-site service contract</td>
</tr>
<tr>
<td></td>
<td>Maintain/store data backups</td>
<td>Maintain/store data backups</td>
</tr>
<tr>
<td></td>
<td>Carry spare batteries, etc.</td>
<td>Documented/tested recovery strategy</td>
</tr>
<tr>
<td>System</td>
<td>Use of any standard features</td>
<td>Use of any standard security features</td>
</tr>
<tr>
<td></td>
<td>Password protection</td>
<td>Password protection</td>
</tr>
<tr>
<td></td>
<td>Virus checking</td>
<td>Virus checking</td>
</tr>
</tbody>
</table>
Methodology for health care data security

Hard disk encryption

Procedural

Stress individual accountability for machine/data when off-site

Personnel

Operational environment

Example countermeasures

Countermeasure category

Site

Use of staff ID badges

Receptionist/guard at main entrance

Room access control (locks)

Alarm systems

Disaster planning

Smoke and moisture detectors

Fire alarm (linked to fire station)

Procedural

Visitors escorted (non-public areas)

Strangers challenged (non-public areas)

Prohibit smoking

Personnel

Controlled access hours

Defined responsibilities

Monitor maintenance work

Data sensitivity

Example countermeasures

Countermeasure level

Disclosure

Denial/destruction

Modification

Medium

File-level passwords

SMART cards

Hard-copy controls

Regular recovery checks

Alternative processing arrangements

Disk shadowing

Resource control

File-level passwords

Integrity checks

Auditing

High

Encrypted transmission

Encrypted storage

Removable storage media

Secure disposal of media/paper

Backup generators

Separation of key assets

Digital signature

4.2. Example 2

4.1.1. Scenario. A pharmacy department serving a large general hospital uses a minicomputer-based system for drug administration. The system may be accessed from a number of locations within the HCE over a local area network.
4.1.2. Methodology implementation

Stage 1: Determine basic system profile
Computer configuration: mini/mainframe; Network—LAN.
Operational environment: location—fixed, urban; building—multiple, modern;
people—staff, public, contract, high.

Stage 2: Determine data sensitivity

Stage 2.1. Identify data groups. Three data groups are encompassed, and can be
identified from the existing model as shown in figure 7.

Stage 2.2: Determine disclosure impacts

<table>
<thead>
<tr>
<th>Data group</th>
<th>Impact level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clinical services</td>
<td>Low</td>
</tr>
<tr>
<td>Finance</td>
<td>Medium</td>
</tr>
<tr>
<td>Library and information services</td>
<td>High</td>
</tr>
</tbody>
</table>

Stage 2.3: Identify data uses. Potential data uses are identified as follows:
(a) operational non-clinical, (b) financial, (c) planning and resource
management.
Methodology for health care data security

Stage 2.4: Determine denial, modification and destruction impacts

<table>
<thead>
<tr>
<th>Data group</th>
<th>Use</th>
<th>Impact levels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Denial</td>
</tr>
<tr>
<td>Clinical Services</td>
<td>Operational non-clinical</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Financial</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Planning and resource management</td>
<td>Low</td>
</tr>
<tr>
<td>Finance</td>
<td>Operational non-clinical</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Financial</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Planning and resource management</td>
<td>Medium</td>
</tr>
<tr>
<td>Library and information services</td>
<td>Operational non-clinical</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Financial</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Planning and resource management</td>
<td>Medium</td>
</tr>
</tbody>
</table>

Stage 2.5: Derive overall sensitivity ratings. The 'worst case' impacts from the previous tables are extracted to determine the overall sensitivity: disclosure, medium; denial, medium; modification, medium; destruction, medium.

Stage 3: Determine appropriate system countermeasures

Computer configuration

<table>
<thead>
<tr>
<th>Countermeasure category</th>
<th>Mini/mainframe Example countermeasures</th>
<th>Network (LAN) Example countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>Control access to computer suite</td>
<td>Protect cabling from interference/tampering (data and power)</td>
</tr>
<tr>
<td></td>
<td>Identifiable marking on terminals</td>
<td>Provide alternate routing</td>
</tr>
<tr>
<td></td>
<td>Site to deny casual access/viewing</td>
<td></td>
</tr>
<tr>
<td>Disaster planning</td>
<td>24-hour maintenance contract</td>
<td>System</td>
</tr>
<tr>
<td></td>
<td>Duplicate/alternative system</td>
<td>Monitor for overuse/failure</td>
</tr>
<tr>
<td></td>
<td>Maintain/store data backups</td>
<td>Automatic re-routing</td>
</tr>
<tr>
<td></td>
<td>Prioritize recovery options</td>
<td>Integrity checking on transmission</td>
</tr>
<tr>
<td></td>
<td>Documented/tested recovery plans</td>
<td>Secure WAN gateways</td>
</tr>
<tr>
<td>System</td>
<td>Use OS security features</td>
<td>Procedural</td>
</tr>
<tr>
<td></td>
<td>Access time/location controls</td>
<td>Maintain list of network assets/access points</td>
</tr>
<tr>
<td></td>
<td>Enforced password criteria</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Automatic terminal logout</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Auditing of activity</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>Log/investigate reported variances</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Control software development/updates</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Formal testing of new programs</td>
<td></td>
</tr>
<tr>
<td>Personnel</td>
<td>Provide software training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Disciplinary procedures for misuse</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Avoid reliance on individuals</td>
<td></td>
</tr>
</tbody>
</table>
Operational environment

<table>
<thead>
<tr>
<th>Countermeasure category</th>
<th>Example countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>Security patrols, Closed-circuit TV monitoring, Use of staff ID badges, Receptionists/guards for sensitive areas, Room access control (locks), Alarm systems</td>
</tr>
<tr>
<td>Disaster planning</td>
<td>Smoke and moisture detectors, Fire alarm (linked to fire station), Backup generator</td>
</tr>
<tr>
<td>Procedural</td>
<td>Visitors escorted (non-public areas), Strangers challenged (non-public areas), Prohibit smoking</td>
</tr>
<tr>
<td>Personnel</td>
<td>Defined responsibilities, Controlled access hours, Monitor maintenance work</td>
</tr>
</tbody>
</table>

Data sensitivity

<table>
<thead>
<tr>
<th>Countermeasure level</th>
<th>Disclosure</th>
<th>Denial/destruction</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>File-level passwords, SMART cards</td>
<td>Regular recovery checks, Alternative processing arrangements</td>
<td>File-level passwords, Integrity checksums</td>
</tr>
<tr>
<td></td>
<td>Hardcopy controls</td>
<td>Resource control, Disk shadowing</td>
<td>Auditing</td>
</tr>
</tbody>
</table>

5. Future enhancement

The most significant extension that is planned is to develop an expert system to be used in conjunction with the methodology. This would contain the expert knowledge necessary to apply the methodology, as well as a knowledge base of appropriate countermeasures.

An expert system would contribute further to the user-friendliness and general accessibility of the method, as it would allow the techniques to be used by health care staff who were not necessarily security-trained (e.g. a hospital general manager). A major advantage of this would be cost, as expensive consultancy would not be required to carry out security reviews. If the system was developed for PC environments it could be made available in nearly all HCE environments.

6. Conclusions

The paper should have served to illustrate how high-level categorizations of health care systems may be used to simplify considerably the process of security selection. Such an approach would be valuable in cases where a full security review has been denied on the grounds of budget or inconvenience.
It is envisaged that the overall methodology should be compatible with the majority of systems, catering for a range of general existing system categorizations. Despite this, however, it is still conceivable that systems will be encountered that do not fit comfortably within the profiles suggested. In these cases it will be necessary to perform a more detailed risk analysis to determine the specific requirements of the system/environment. Additionally, in systems where extremely high levels of risk are identified, more detailed study is also advisable.

The methodology itself is at an early stage of development, and requires further refinement before it can be considered practically viable. The next stage of development will be to encompass it within an expert system so that it can be used within various HCE environments. This will serve to test the methodology and allow adjustments to be made accordingly.

Acknowledgements
We would like to acknowledge the various partners and collaborators within the SEISMED project for their contributions to the content of this paper.

References
AN EXPERT SYSTEM FOR HEALTH CARE DATA SECURITY:
A CONCEPTUAL OUTLINE

S M Furnell, P W Sanders and C T Stockel

Network Research Group, Faculty of Technology, University of Plymouth,
United Kingdom.

ABSTRACT

Information systems security is now an important consideration in modern health care establishments (HCEs), given their increased reliance on information technology in both direct care and administration activities. The paper outlines the basic framework and functions of an expert system tool to assist with the specification and selection of security countermeasures in HCEs. The discussion is based upon a generic protection methodology that has been developed as a means of categorising existing medical information systems according to pre-determined protection "profiles! and identifying their security requirements. It is envisaged that the incorporation of this method within an expert system framework could potentially enhance countermeasure selection and allow requirements to be established by non-professionals.

The content is based upon work initiated as part of the Commission of European Communities SEISMED (Secure Environment for Information Systems in MEDicine) project, the aim of which is to provide security recommendations for European Health Care Establishments (HCEs).

INTRODUCTION

Modern Health Care Establishments (HCEs) now place a great reliance upon information technology, and contain a large number of systems processing many types of health data [1]. In many cases the information held is directly related to patients and care delivery and can, therefore, be of a sensitive nature. In addition, much information exists that is vital to the smooth operation of the HCE in general. These considerations dictate an
obvious need for data security within the environment, which is not present in many existing systems [2].

The objective of information security is to protect all aspects of the computing system (e.g. the computer itself, the building in which it is housed and the data that is stored). At the highest level, the security of information systems dictates the following requirements:

- confidentiality (i.e. information is only disclosed to authorised users);
- integrity (i.e. information can only be modified by authorised users);
- availability (i.e. information and other IT resources can be accessed whenever needed).

Health care computing systems provide many potential targets for a security breach, complicating the protection issue. The myriad of different system configurations and types of information render it impossible to assert a single level of security that will be appropriate for all cases.

As a result, individual HCEs must determine the level of security that is appropriate for their environment, as well as where and how measures should be introduced. The standard approach is to conduct a risk analysis investigation, often using outside consultants. Unfortunately, this can be a time consuming and costly proposition and may consequently be prohibitive in many cases, resulting in the potential compromise of security. However, given that many threats and vulnerabilities are not unique to individual HCEs, a full risk analysis in each case may also be largely unnecessary.

This paper begins by introducing a methodology that has been developed to simplify the identification of security requirements for existing systems. It then proceeds to consider the potential for implementing these concepts within an expert system framework, and the advantages that this would provide.

A GENERIC PROTECTION METHODOLOGY

A methodology has been developed that enables security requirements of existing information systems to be established by analysing the following key elements:

1. computer configuration;
2. operational environment;
3. data sensitivity.

A number of pre-determined security profiles have been devised for each system element, and using appropriate combinations of these it is possible to specify suitable high level system profiles to describe the majority of application areas.

The computer configuration and operational environment elements are considered to determine the baseline security requirements of a system. Consideration of data sensitivity builds upon this, examining the additional security measures demanded by the nature of the data involved.
The key information system elements recognised by the methodology will now be described in more detail.

- **Computer Configuration**
  This refers to the IT assets (both hardware and software) of the organisation. At a high level it is possible to identify a relatively small number of elements which may be included in any given computer configuration:

  - mini-computer;
  - desktop PC;
  - LAN;
  - mainframe;
  - portable / laptop;
  - WAN.

  Individual systems would be considered to determine which elements are applicable, and associated countermeasures selected accordingly.

- **Operational Environment**
  The nature of the environment in which IT assets are located and used will also affect the type and level of protection required. The main environmental considerations that have security bearing are:

  - location (e.g. fixed / mobile; rural / urban / city);
  - buildings (e.g. single / multiple; old / modern);
  - people (e.g. low / medium / high number; staff / contract / general public);

  Appropriate combinations of these factors can be used to describe the majority of health care establishments (i.e. from GPs to general hospitals). Security countermeasures have again been specified for each type of environment.

- **Data Sensitivity**
  Sensitivity is determined by the *types* and *uses* of data within a system. In consultation with a number of European HCEs, a generic model of medical data was developed [3], providing a high level means of specifying the types of data utilised within any system. The model comprises 12 main data groups and applications may use information from all groups, or a particular subset. The issue of data use was addressed by related work within the SEISMED project [4], which determined a similarly high level set of 9 generic data uses that are compatible with the model. The data types and uses are listed in table 1.

<table>
<thead>
<tr>
<th>DATA TYPES</th>
<th>DATA USES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Identification</td>
<td>Operational Clinical</td>
</tr>
<tr>
<td>Patient Administration</td>
<td>Emergency Care</td>
</tr>
<tr>
<td>Patient Care</td>
<td>Critical Clinical</td>
</tr>
<tr>
<td>Clinical Services</td>
<td>Expert Systems</td>
</tr>
<tr>
<td>Finance</td>
<td>Operational non-clinical</td>
</tr>
<tr>
<td>Hotel Services</td>
<td>Financial</td>
</tr>
<tr>
<td>Staff</td>
<td>Planning &amp; Resource Management</td>
</tr>
<tr>
<td>Resource Management &amp; Planning</td>
<td>Quality Management</td>
</tr>
<tr>
<td>Library &amp; Information Services</td>
<td>Clinical Research</td>
</tr>
<tr>
<td>Expert Systems</td>
<td></td>
</tr>
<tr>
<td>Communication Services</td>
<td></td>
</tr>
<tr>
<td>External Systems</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Generic Data Types and Uses
Sensitivity is rated by considering four types of potential impact related to the data types and uses identified in the system, as listed in table 2.

<table>
<thead>
<tr>
<th>IMPACT TYPE</th>
<th>DEFINITION</th>
<th>DERIVED FROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disclosure</td>
<td>Unauthorised disclosure of data to HCE staff or outsiders</td>
<td>Data type</td>
</tr>
<tr>
<td>Denial</td>
<td>Denial of access to data for varying periods</td>
<td>Data use</td>
</tr>
<tr>
<td>Modification</td>
<td>Accidental or deliberate alteration of the data</td>
<td>Data use</td>
</tr>
<tr>
<td>Destruction</td>
<td>Destruction of the system or data (an extreme form of denial)</td>
<td>Data use</td>
</tr>
</tbody>
</table>

Table 2: Data impacts

The impacts are rated low, medium or high (where low indicates that the baseline countermeasure level is satisfactory and high is the maximum protection that can be provided). The level in each case was determined by considering potential threats to the HCE that may result (e.g. breach of confidentiality, embarrassment, disruption of activity, financial loss, legal action, threats to personal safety), with opinions being gathered from various European HCEs. Example ratings are given in table 3.

<table>
<thead>
<tr>
<th>DATA TYPE / USE</th>
<th>IMPACT TYPE</th>
<th>RATING</th>
<th>REASON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Care</td>
<td>Disclosure</td>
<td>High</td>
<td>confidentiality, embarrassment, legal</td>
</tr>
<tr>
<td>Operational Clinical</td>
<td>Denial</td>
<td>Medium</td>
<td>disruption, safety</td>
</tr>
<tr>
<td></td>
<td>Modification</td>
<td>High</td>
<td>safety, legal</td>
</tr>
<tr>
<td></td>
<td>Destruction</td>
<td>Medium</td>
<td>disruption, safety</td>
</tr>
</tbody>
</table>

Table 3: Sensitivity ratings

The level of impact will in turn determine the level of countermeasure (with the countermeasures used being derived from a representative set that are being developed for use within the SEISMED project [5]).

These elements have been incorporated into the methodology framework as shown in figure 1, illustrating (at a high level) the steps involved in profiling the security requirements of existing systems.

Fig 1: Existing System Protection Methodology Overview
A more comprehensive description of the methodology and its application can be found in [6].

**EXPERT SYSTEM IMPLEMENTATION**

This section describes the potential for implementing the methodology as an expert system.

The main purpose would be to provide an intelligent decision support tool to assist in applying the methodology. It is considered that the most appropriate approach would be for the system to be based around a "consultation" style of interaction, guiding the user through each stage of the security analysis process.

The expert system knowledge base would contain a full range of countermeasures and selection rules, based upon the existing set associated with the methodology and additional expertise gathered from security consultants to enable further inferences (the latter would related more to selection rules than actual countermeasures, being based upon the experts own experiences).

The principal stages of the expert system analysis would correspond closely to the normal steps in applying the methodology. The system would need to elicit a fundamental system description from the user by identifying the following factors:

- computer configuration involved;
- type of operational environment(s);
- data groups involved;
- purpose of application (data use(s)).

The majority of the user interaction would occur at this stage, with the system querying the user to establish which elements are present. The level of expertise employed would be dependent upon the security and IT experience of the user. Therefore, in the case of the analysis being driven by a relatively novice user, the system would rely upon a detailed style of consultation in order to elicit the required knowledge. Conversely, experienced staff would be more likely to utilise the system as an automated methodology tool.

An initial system profile would be derived from the consultation using a series of basic selection rules associated with each methodology category, for example:

```
IF computer configuration includes Desktop PC
THEN countermeasure 1
    countermeasure 2 etc.

IF computer configuration includes LAN
THEN countermeasure 3
    countermeasure 4 etc.
```
At this stage countermeasures could be extracted directly from the knowledge base without any need for further inference.

The basic profile would not take into account any practical constraints that may apply with regard to countermeasure implementation (e.g. financial limitations, operational constraints and/or existing countermeasures). Further consultation to establish such constraints could be used as the basis for filtering of the countermeasure suggestions (as represented by the transition from appropriate to selected countermeasures in figure 1). This would, however, demand that the data in the knowledge contained information about both “implementation difficulty” and costs (the latter of which would need to be updated regularly in order to be practical).

Having established the basic profile and any constraints, more advanced selection rules could be utilised to allow inferences based upon information from across several categories (which would be based upon the additional knowledge gathered from the experts). This would potentially allow the identification of additional requirements that may have been missed during the initial consultation. Examples of these further rules may be as follows:

IF LAN AND High Disclosure rating
THEN transmission encryption.

IF Minicomputer AND High Integrity rating
THEN file checksums.

It is anticipated that the overall structure of the system, and the process of user interaction involved, would be as illustrated in figure 2.
It is envisaged that the use of expert system technology would provide a number of advantages. At the most basic level it would serve to ensure the correct and consistent application of the methodology concepts.

However, the encapsulation within this framework would offer opportunities beyond the simple automation of the methodology. Having established the basic system profile by following through the key methodology stages, the specification could then be enhanced using inferences based upon the advanced rules in the knowledge base. The countermeasure recommendations would then be narrowed, making them more specific to the system under consideration.

Finally, an expert system would improve the user friendliness and general accessibility of the method. It would offer a significant opportunity for the techniques to be employed by health care staff who were not necessarily fully security-trained (e.g. the hospital IT manager). A major advantage of this would be reduced cost, eliminating the need for the expensive consultancy normally involved in carrying out security reviews and allowing them to be conducted "in-house". If such a system were to be developed for the PC environment then this would guarantee the maximum potential for adoption, given that this platform is available in nearly all HCE environments.

CONCLUSIONS

The issue of information security cannot be ignored within the health care field. The use of a specifically tailored methodology as described will simplify the security selection process and would be valuable in cases where a full security review may have been denied.

The discussion of the potential for expert system implementation highlights additional benefits that may result, and further extends the scope of the methodology. However, it must be acknowledged that any advantages are theoretical at this stage, and that the expert system design must be trailed in practice to provide any empirical evidence of its true worth.

The development of an actual system (in conjunction with further refinements to the underlying methodology) is viewed as the next stage in the evolution of the concept.

REFERENCES

THE USE OF KEYSTROKE ANALYSIS FOR CONTINUOUS USER IDENTITY VERIFICATION AND SUPERVISION

Steven M Furnell, Peter W Sanders and Colin T Stockel
Network Research Group
Faculty of Technology
University of Plymouth
Plymouth, United Kingdom
E-mail: stevef@soc.plym.ac.uk

ABSTRACT

This paper proposes the concept of dynamic keystroke analysis as a means of enhancing user authentication in modern information systems. Whilst existing password-based schemes normally rely upon a single authentication judgement, the use of keystroke analysis would allow supervision to occur continuously throughout user sessions. In addition, the concept may be implemented transparently so as not to unnecessarily disrupt user activity. These points make it suitable for application in modern, user-friendly contexts such as multimedia.

The theoretical discussion is supported by the findings of an experimental study mounted within our group using 26 typists and a prototype authentication system. The results demonstrate considerable success, with an impostor detection rate of 85%. However, a number of potential problems identified in the discussion suggest that keystroke analysis would be best implemented alongside other supervision techniques rather than as a standalone system.

INTRODUCTION

A key issue in the implementation of secure information systems is user authentication. The password remains the popular and widespread technique (National Computing Centre 1994), having the advantage of simplicity for both systems designers and end users. However, a disadvantage is the ease with which its protection is often compromised, either deliberately, by accident or by guesswork. In recent years the reliability of passwords has been repeatedly questioned (Jobusch and Oldehoeft 1989) and it is now widely accepted that stronger means of authentication may be necessary, using techniques that are more difficult to forge. In addition, password techniques can only verify user identity at discrete points within a session (and are normally only incorporated at the beginning). With the increasing advancement of information systems, as witnessed by the progression to multimedia, it is both desirable and appropriate to have a means of identity verification that can deliver a continuous assessment of user legitimacy (and thereby provide greater protection against compromise).

This paper proposes a behavioural biometric measurement based upon the analysis of users typing characteristics. It has been established that users may exhibit significant differences in terms of typing styles and abilities (Card et al. 1980), which may consequently be used to determine reasonably unique typing "signatures" (analogous to those which can be identified with normal handwriting (Fairhurst et al. 1994)). These signatures may then be used as the basis for real-time user supervision, providing a continuous and transparent (i.e. non-intrusive) means of verifying user identities in conjunction with their normal working activities.

CONCEPTUAL SUMMARY

Several typing characteristics may be considered as the basis for determining keystroke signatures, including the intervals between keystrokes, the duration of keystrokes and the frequency of typing errors. The chosen factors must be assessed to create a typing profile for each legitimate user. Subsequent authentication / supervision is then based upon a comparison of the current users typing characteristics against the profile associated with his / her claimed identity (with any significant departures triggering impostor alerts).

Keystroke analysis may be implemented in two ways (referred to as static and dynamic verification approaches), which differ in how they attempt to use the technique. In the static scenario, analysis is based upon a constant text string and is normally used for a single authentication judgement (e.g. in conjunction with the entry of a normal user id and password). By contrast, the dynamic approach attempts to analyse any arbitrary text input, allowing much greater scope for user supervision as the authentication period may become continuous. The majority of previous studies have concentrated upon the static verification approach (Bleha et al. 1990; Joyce and Gupta 1990).
As with other biometric systems, the effectiveness of keystroke analysis is judged on the basis of two factors:

- **False Acceptance Rate (FAR)**
  The proportion of cases in which impostors are falsely authenticated by the system (also referred to as Impostor Pass Rate).

- **False Rejection Rate (FRR)**
  The proportion of cases in which legitimate users are rejected by the system (also referred to as False Alarm Rate).

Acceptable figures for these measures are heavily dependant upon whether a static or dynamic verification strategy is employed. In the static scenario, minimising the FAR should be the most important consideration, as any successful impostor could potentially go unchecked for a whole session. However, in the dynamic scenario, with continuous assessment, a greater window for impostor detection is available and so the prime concern becomes to minimise the FRR (as rejections during a session could irritate and disrupt a legitimate user more significantly than occasional false login failures).

**PRACTICAL STUDY**

This section details the research team's implementation of a prototype keystroke authentication system based on the dynamic verification approach.

**Experimental System**

An experimental system has been developed for the PC environment to allow an evaluation of the concept in practice. It is comprised of three modules, as follows:

- **Profiler**
  Accepts the initial typing samples used to create profiles for legitimate users. PC hardware interrupts are used to detect key depression and release with one millisecond accuracy.

- **Sampler**
  Accepts user test samples and stores all keystrokes and associated timings for later use.

- **Monitor**
  Compares the test samples against typing profiles to determine the effectiveness of the system.

Typing profiles were based upon inter-keystroke times for specific character pairs (digraphs), storing the mean time and standard deviation for each profiled digraph (note: inter-keystroke time was found to be the most distinctive typing characteristic in a provisional study, with the keystroke duration and typing error frequency measures exhibiting FARS significantly high enough to warrant exclusion from further investigation). Analysis was restricted to digraphs involving alphabetic and "space" characters, as these were considered the most likely to reveal any characteristic keystroke rhythm and were found to produce the best results in a previous study which conducted a comprehensive investigation of this aspect (Leggett and Williams 1988).

The profiling procedure demanded that users enter two samples of a 2200 character *reference text*. A significant length was necessary to ensure that each users "natural" typing style emerged and that sufficient samples of each digraph were obtained to enable appropriate mean and standard deviation values to be established (note that at least five samples were required for profile entries to be usable in monitoring, as any less could result in them being unrepresentative of the users normal style). Another property of the reference text was that the relative frequencies of character digraphs within it corresponded closely to those of normal English (Beker and Piper 1982), with the 30 most common digraphs all significantly represented (ensuring strong profile entries for the digraphs most likely to be encountered).

The profiler attempted to further ensure representative profiles by filtering out potentially uncharacteristic typing. This was achieved in two ways: firstly, deleted keystrokes were ignored, as any entries resulting from mis-strokes could be unrepresentative. Secondly, inter-keystroke times exceeding 750ms (i.e. Card et al's speed classification for a user unfamiliar with the keyboard) were disregarded, being considered more likely to represent unnatural pauses than part of the users typing rhythm.

The monitoring / supervision system compared incoming inter-keystroke times (from the test samples) against user profiles, with times being judged invalid if they fell outside 1.5 standard deviations of the relevant profiled value. Invalid keystrokes were then analysed in two ways to detect intrusions:

1. monitoring the percentage of invalid keystrokes during the 100 most recently typed;
2. monitoring the number of consecutive invalid keystrokes.

However, even legitimate users will generate some invalid keystrokes and, as a result, the monitor incorporates user-specific authentication thresholds.
which specify the maximum levels for percentage invalid keystrokes and consecutive invalid keystrokes that are tolerated against each profile (note that the use of common threshold levels for all users was found to be less effective). The appropriate levels were determined using the two further text samples (of 574 and 389 characters) entered by each user, which were run against their initial profile. The peak values observed for percentage invalid keystrokes and consecutive invalid keystrokes across the two tests were then used as the basis for establishing the thresholds. If either threshold was exceeded during monitoring, an intrusion alert was generated.

Given that the dynamic verification approach was being tested, minimising the FRR was considered important. The user-specific thresholds were, therefore, set to ensure that no false rejections would arise from the test samples. The advantage of this was that the resulting FAR would then effectively represent a "worst case" figure.

Test Subjects

A total of 26 subjects were involved in the tests, with abilities ranging from experienced typists to comparative novices.

The two additional text samples that had been used to determine the authentication thresholds for legitimate users were also used to represent impostor attempts (by running them against all other profiles). The final results were, therefore, derived from approximately 1300 impostor attempts.

Results and Analysis

With the FRR having been eliminated, the aims of the study were to determine the FAR and the speed of successful impostor detection.

In terms of overall impostor detection effectiveness, the experimental system exhibited a FAR of 15%, as shown in figure 1. However, given that each subject provided two test samples, it was also possible to investigate impostor consistency. This was established by subdividing the test samples into the pairs that were generated by the same subjects and then determining the proportion of cases where both samples were able to pass as another user against those where only one attempt was successful. This information is also illustrated in figure 1. It can be conjectured that, given longer test samples, the impostors who were successful in only one attempt would eventually be detected at some point (albeit after a significant number of keystrokes) and that the FAR would, therefore, be somewhat less.

The performance of the two detection methods employed was found to be very similar, with 49% of impostors being detected as a result of their percentage of invalid keystrokes, against 51% due to consecutive invalid keystrokes. As such, both methods can be considered to be useful authentication measures.

Given that impostor detection is actually possible, the next most important consideration is the speed with which it can be achieved (i.e. how many keystrokes would an impostor be able to enter before being detected - a factor which does not appear to have been addressed in previous studies). The experimental findings on this aspect are shown in figure 2 below. This shows the percentage of impostors detected within five distinct keystroke ranges, with cumulative values also indicated.
These results indicate that the vast majority of impostors would be detected within 160 keystrokes (the equivalent of 2 standard lines of text), with detection in under 40 keystrokes in 25% of cases. Whilst this may not combat the most destructive scenarios (e.g. the immediate entry of "delete *.*" would very likely be unchallenged), it should be sufficient to identify the more common types of intruder who generally require sustained access in order to effect a serious breach.

It should also be noted that these figures essentially characterise the impostor detection performance that would be observed from the point of initial login (i.e. beginning with 0% invalid keystrokes). However, in scenarios where an impostor takes the place of a legitimate user it is likely that detection would be quicker and more frequent, as a certain percentage of invalid keystrokes would already have been registered (by the legitimate user) and, therefore, the rejection threshold would be reached more easily.

A FAR of 15% would be of less significance if the preliminary user identification phase was still to include some other form of authentication as well (e.g. a standard password system) as the combination would almost certainly serve to foil the majority of intrusion attempts.

A FRR of 0% is of course somewhat artificial, as some false rejections would be almost bound to occur in practice. However, with authentication thresholds set correctly, it is envisaged that these cases would not be frequent enough to significantly trouble legitimate users.

ADVANTAGES OF KEYSTROKE ANALYSIS

The principal advantages of the approach are improved security, reduced cost and user convenience - some of which cannot be claimed for many alternative authentication methods.

Improved security is advantageous in any information system, and is achieved here as authentication is no longer restricted to a single judgement, but may become continuous throughout the session. In addition, the biometric nature of the approach makes it more difficult for users themselves to undermine security (e.g. by allowing colleagues unauthorised access to their accounts) as typing abilities cannot be passed on to someone else in the same way as a password.

Cost advantages result from the fact that it is possible to implement the concept entirely in software (with the necessary recognition hardware already present in the form of existing PCs), whereas many frequently suggested authentication enhancement schemes (e.g. Smart cards, other biometric methods) are reliant upon specialised equipment. This makes the technique particularly suited to financially constrained environments. Cost may also be an important consideration in multimedia systems, as these require expensive base technologies which may leave little scope for additional expenditure on security.

Finally, user convenience comes from the fact that identity verification can be performed transparently, in a non-intrusive manner. This is an important consideration, particularly in a multimedia context, and is illustrated in figure 3. This shows a potential means of implementing keystroke analysis, with the existence of the monitor remaining transparent to the user unless an intrusion is suspected.

ADVANTAGES OF KEYSTROKE ANALYSIS

The principal advantages of the approach are improved security, reduced cost and user convenience - some of which cannot be claimed for many alternative authentication methods.

Improved security is advantageous in any information system, and is achieved here as authentication is no longer restricted to a single judgement, but may become continuous throughout the session. In addition, the biometric nature of the approach makes it more difficult for users themselves to undermine security (e.g. by allowing colleagues unauthorised access to their accounts) as typing abilities cannot be passed on to someone else in the same way as a password.

Cost advantages result from the fact that it is possible to implement the concept entirely in software (with the necessary recognition hardware already present in the form of existing PCs), whereas many frequently suggested authentication enhancement schemes (e.g. Smart cards, other biometric methods) are reliant upon specialised equipment. This makes the technique particularly suited to financially constrained environments. Cost may also be an important consideration in multimedia systems, as these require expensive base technologies which may leave little scope for additional expenditure on security.

Finally, user convenience comes from the fact that identity verification can be performed transparently, in a non-intrusive manner. This is an important consideration, particularly in a multimedia context, and is illustrated in figure 3. This shows a potential means of implementing keystroke analysis, with the existence of the monitor remaining transparent to the user unless an intrusion is suspected.

ADVANTAGES OF KEYSTROKE ANALYSIS

The principal advantages of the approach are improved security, reduced cost and user convenience - some of which cannot be claimed for many alternative authentication methods.

Improved security is advantageous in any information system, and is achieved here as authentication is no longer restricted to a single judgement, but may become continuous throughout the session. In addition, the biometric nature of the approach makes it more difficult for users themselves to undermine security (e.g. by allowing colleagues unauthorised access to their accounts) as typing abilities cannot be passed on to someone else in the same way as a password.

Cost advantages result from the fact that it is possible to implement the concept entirely in software (with the necessary recognition hardware already present in the form of existing PCs), whereas many frequently suggested authentication enhancement schemes (e.g. Smart cards, other biometric methods) are reliant upon specialised equipment. This makes the technique particularly suited to financially constrained environments. Cost may also be an important consideration in multimedia systems, as these require expensive base technologies which may leave little scope for additional expenditure on security.

Finally, user convenience comes from the fact that identity verification can be performed transparently, in a non-intrusive manner. This is an important consideration, particularly in a multimedia context, and is illustrated in figure 3. This shows a potential means of implementing keystroke analysis, with the existence of the monitor remaining transparent to the user unless an intrusion is suspected.

ADVANTAGES OF KEYSTROKE ANALYSIS

The principal advantages of the approach are improved security, reduced cost and user convenience - some of which cannot be claimed for many alternative authentication methods.

Improved security is advantageous in any information system, and is achieved here as authentication is no longer restricted to a single judgement, but may become continuous throughout the session. In addition, the biometric nature of the approach makes it more difficult for users themselves to undermine security (e.g. by allowing colleagues unauthorised access to their accounts) as typing abilities cannot be passed on to someone else in the same way as a password.

Cost advantages result from the fact that it is possible to implement the concept entirely in software (with the necessary recognition hardware already present in the form of existing PCs), whereas many frequently suggested authentication enhancement schemes (e.g. Smart cards, other biometric methods) are reliant upon specialised equipment. This makes the technique particularly suited to financially constrained environments. Cost may also be an important consideration in multimedia systems, as these require expensive base technologies which may leave little scope for additional expenditure on security.

Finally, user convenience comes from the fact that identity verification can be performed transparently, in a non-intrusive manner. This is an important consideration, particularly in a multimedia context, and is illustrated in figure 3. This shows a potential means of implementing keystroke analysis, with the existence of the monitor remaining transparent to the user unless an intrusion is suspected.
- **Consistency of users**
  Users typing performance may be adversely affected by many factors (e.g. fatigue, injury, keyboard variations, interruptions), leading to departures from their profiled level.

- **Mimicry**
  It may be possible for impostors to deliberately imitate the keystroke "signatures" of legitimate users (particularly poor typists).

- **Timing accuracy**
  The concept can only be implemented in networked environments if accurate inter-keystroke timings can be obtained at the local terminals.

- **User acceptance**
  Some users may object to the idea of their activities being continuously monitored, leading to potential resistance.

- **General applicability**
  A keyboard-intensive context is required if monitoring is to be effective. However, some applications (particularly in multimedia) significantly reduce the role of the keyboard.

It would be possible to compensate for this last point by specifically profiling and monitoring words or key combinations that are still known to be frequently typed (thereby applying a static verifier approach in a dynamic context).

**CONCLUSIONS**

We believe that the experimental study has served to confirm the significant potential of keystroke analysis as a means of user supervision. Whilst it cannot be regarded as a panacea to the authentication issue, it should, at the very least, provide considerable protection over the use of a simple password alone.

The experimental system is currently being enhanced to enable more extensive investigation. Firstly, a full implementation of the system has been developed that runs transparently on a user workstation. In this scenario keystroke data is collected locally and then analyzed by a monitoring system operating on another machine. Secondly, neural network techniques are being incorporated to allow the system to learn how best to conduct its analysis (for example, to determine which character digraphs are the most distinctive for a particular user). Once these enhancements have been completed, the resulting system will provide a much better indication of the concepts real-world potential.

It is considered that the FAR could be reduced by generating more representative profiles of legitimate users. Whilst this would require larger text samples (which could be collected via a background process to reduce the user burden), it would potentially allow more accurate authentication thresholds to be set and reduce the number of unrepresented digraphs in the profiles (therefore allowing more keystrokes to be analyzed).

Despite this, it is felt that keystroke analysis would be most effectively used in conjunction with other forms of supervision, as a supplementary means of authentication (with passwords, or some other appropriate technique, still being employed as the primary mechanism). This would provide an opportunity to combat the FAR and could also reduce the significance of the potential problems identified above. As such, the eventual aim of the research is to incorporate the concepts into a more comprehensive intrusion monitoring framework, using a number of additional behaviour parameters to identify departures from normal system usage.

**REFERENCES**


SECURE MULTIMEDIA SYSTEMS IN HEALTHCARE AND MEDICINE

Steven M Furnell, Nichola J Salmons, Peter W Sanders, Colin T Stockel and Matthew J Warren
Network Research Group
Faculty of Technology
University of Plymouth
Plymouth, United Kingdom
E-mail: nichola@soc.plym.ac.uk

ABSTRACT

The aim of this paper is to examine the increasing potential for applying multimedia technology within the medical community. Multimedia is considered to be a particularly appropriate means for information delivery within Healthcare Establishments (HCEs), especially for that relating to patient care, and the paper considers the principal advantages in this area. The discussion then proceeds to highlight the fact that adoption of multimedia dictates new requirements for information security and, by the nature of the technology involved, also allows new approaches to be explored. On this premise, the outline of a security strategy for future multimedia healthcare networks is proposed. The discussion is supported by an example scenario and a brief examination of our own research groups efforts in this area.

INTRODUCTION - MULTIMEDIA IN MODERN MEDICAL CARE

Over the past twenty years computerised information systems have gradually been introduced to, and utilised within, a large number of healthcare establishments (HCEs). Information Technology (IT) now enables modern HCEs to provide more comprehensive medical care, comprising more numerous and more complex procedures. As such, HCE systems now process and handle information beyond simple text and graphics and more advanced medical applications may also generate digital images, full motion video and audio. The use of this multimedia information can considerably aid patient diagnosis and treatment (Ceusters et al. 1993).

As a result of recent advances in desktop processing power, the large scale use of multimedia-based healthcare systems is closer to being an achievable goal, with the presentation and delivery of multimedia information becoming possible at a viable price. This is largely due to the fact that PC-based systems can now represent a realistic platform for multimedia and can be found in numbers in most HCEs. In addition, telecommunications networks are now capable of handling the high speeds necessary to transfer large amounts of multimedia data, allowing further improvements to the speed of information delivery within and between HCEs.

In terms of advantages, the presentation of medical data in a multimedia format is considered to be ideally suited to the healthcare field as it inherently provides more information (Orozco-Barbosa et al. 1992), and in a form that is more easily comprehended than traditional text-based reports. This should indirectly help to improve the quality of care, as clinical decisions are made on the basis that the clinician has direct access to the most comprehensive information possible. In addition, it will allow the seamless integration of existing operational systems, with the ability to maintain a standardised viewing structure. As such, the potential applications of multimedia in healthcare are wide-ranging. For example, an area of significant potential will be the establishment of composite electronic health records, bringing together various types of multimedia patient data into a single entity (Arnold and Peter 1993). Such electronic multimedia record systems have the potential to significantly improve care delivery as they will allow immediate access to full patient data at any time, with flexible options for retrieval (whereas the same data may currently be held in several different places, making it difficult for clinicians to obtain all of the information that may be available).

REQUIREMENTS FOR SECURITY

It is important to recognise that a major consequence of the progression to multimedia will be an extension of the already significant reliance upon IT in healthcare establishments. This reliance stems from the increasing number of healthcare IT applications, particularly those relating to clinical care, that are now fundamental to routine clinical practice (Barber 1991). A number of future trends are predicted (European Commission 1994), with European project sponsorship (in the 4th Framework) under way, that will further increase this dependency. These include:

- increased intra and inter-HCE networking:
- increased exchange of data between HCEs;
- increased potential for sharing of facilities between HCEs;
- establishment and adoption of the composite electronic health record.

Due to the comprehensive nature of the information presented, it is envisaged that there is likely to be a even greater level of implicit trust in the correctness of the system. As such reliance upon IT increases, so too does the potential impact of any system unavailability or erroneous data. This, therefore, heightens the requirement to ensure that the availability and integrity of medical systems can be maintained.

In addition, further considerations arising from the increasing variety and complexity of data dictate a greater need for confidentiality controls. Firstly, the amalgamation of different forms of data into the composite record may potentially increase the sensitivity of the information beyond that of any of the component parts. Secondly, information that would previously have been held (and potentially secured) by separate applications would now be placed together, and thus the impact of a security breach would be significantly higher. The use of multimedia can, therefore, be seen to affect all three main principles of information security (i.e. confidentiality, integrity and availability).

As a result of these considerations, the authors believe that a different approach may be necessary to integrate security into multimedia systems and that the environment may also allow new opportunities to be explored.

A SECURITY STRATEGY FOR MULTIMEDIA HEALTHCARE SYSTEMS

Whilst many areas of security (e.g. physical, environmental and personnel considerations) will not be directly affected by the multimedia context, there will be noticeable effects in others; some significant, some less so (e.g. the quantity of data involved will affect the backup process in terms of increased storage requirements and, potentially, the time required to perform the task). The paper concentrates upon two aspects in particular which should be re-examined in light of the trends predicted above; namely user authentication and data communications. In both of these cases, an important issue will be the transparency of protection mechanisms employed. One of the main advantages of multimedia systems is that data can be presented in a more natural and "user-friendly" context. As such, there is a dilemma that whilst the systems must be easy to use and effective, they must at the same time be made secure. This does not necessarily mean that users should be totally unaware of security (indeed, it will probably increase trust in the system if some security is seen to be present), but it must not interfere with their work and should be compatible with the general "feel" of the system.

User Authentication

User authentication mechanisms will still be required to prevent impostors masquerading at local terminals and workstations. However, two factors suggest that traditional password-based methods alone will no longer be sufficient protection:

- multimedia systems will significantly reduce the role of keyboard input in some contexts (e.g. information retrieval), such that it may not be required at all HCE terminals. As having to retain a keyboard simply for user authentication purposes would hardly constitute transparent security, an authentication mechanism not requiring this aspect would be desirable;
- the increased data sensitivity that could potentially result from the composite record context adds weight to the argument that passwords (which often provide a weak / unreliable basis for authentication anyway (Jobusch and Oldehoeft 1989)) should be supplemented by other mechanisms.

The use of smart card systems may have a place in overcoming these problems, but may not be practical as a compulsory measure as this would introduce an immediate financial burden across the whole system (which most HCEs would not be able to tolerate at the present time).

A appropriate alternative would be to utilise advanced user supervision systems which could operate transparently and in real-time throughout each session (Lunt 1993). A number of factors could potentially be encompassed by the supervision, including:

- times and locations of system usage;
- typical applications used;
- types of data accessed and how it is used;
- analysis of the users typing style (if a keyboard is still used).

The use of neural network techniques could allow appropriate information on these (and other factors) to be gathered automatically, with subtle behaviour patterns
being learnt in order to develop profiles for legitimate system users. Current user activity could then be continuously compared against the profile for the users claimed identity (with significant departures causing an alert to be generated).

In addition to the above, multimedia systems may allow many new options to be introduced for improving authentication. For example, appropriate hardware for implementing several biometric identification methods may already be present "as standard" in a multimedia configuration (e.g. cameras which may be used for image / "faceprint" recognition, microphones and audio processing facilities for voice recognition). These techniques have been successfully implemented elsewhere, delivering adequate authentication performance and gaining a high degree of user acceptance (Sherman 1992). As such they should integrate well with multimedia systems. However, the presence of such hardware enhancements should not be a prerequisite of the authentication strategy for the same reasons as smart cards. Nevertheless, some mechanism should be incorporated to allow extra facilities to be utilised if they are present.

Future multimedia systems may, therefore, demand that a variety of authentication technologies are actually employed, based around an approach that is primarily software-oriented. These may then be linked / managed by an intelligent supervision system which can select the most appropriate mechanism to be invoked at any given point according to the current user activity and the type of system being used (e.g. keystroke analysis could be used in any text-intensive activity; facial recognition could be used if the host system is equipped with a camera). Note that once authentication has been conducted, any underlying data / application access and auditing controls could still be implemented in a traditional manner to restrict and monitor the activities of different classes of user.

Data Communications

One of the trends likely to result from the availability of more and better information is the increased sharing and exchange of data between HCEs. In the UK, the National Health Service (NHS) already plans to bring all aspects of voice and data communications together into a common framework, with all major HCEs having the facility to communicate electronically by 1996 (NHS Management Executive 1992). However, the transmission of composite records again raises the concerns of confidentiality and integrity (i.e. the need to protect messages against unauthorised interception, modification and falsification). Hence the requirement to have secure data communications will also be correspondingly greater. A strategy is proposed that would introduce layered security at local, national and international levels with encryption of data between different security domains (based upon a Trusted Third Party (TTP) approach as shown in figure 1).

![Fig. 1: Secure Data Communications using a TTP hierarchy](image)

The TTP would be capable of providing three main types of security service in relation to data transmission:

- integrity (e.g. checksums);
- non-repudiation (e.g. digital signatures);
- confidentiality (e.g. encryption).

These services would be applied, as appropriate, to communications at all levels of the TTP hierarchy. In addition, encryption could be used to protect stored data where workstations in the local domains cannot be physically secured. However, it should be noted that whilst the facility for encryption would exist, its use in healthcare is currently restricted in some EC countries. The operation of all data communications services could theoretically be made completely transparent to the end user (although in some cases, such as the use of digital signatures, users should be given some indication that a security service is being provided).

As can be seen from the figure, the Security Management Centre (SMC) introduced to handle the authentication system will also assume responsibility for securing communications in each local domain. The SMC facilities could be incorporated as part of an overall Network Management Centre.

This strategy would increase the importance of maintaining availability, with a reliance upon the availability of interconnected systems as opposed to earlier isolated ones. The hierarchy would, therefore, be
designed to be fault tolerant to enable secure operations to continue even in the event of individual TTP failure.

However, this strategy obviously depends upon the overall TTP infrastructure being in place before it can be realised. Therefore, in the short to medium term, individual HCEs and co-operating establishments will require alternative means by which their communications can be secured (AIM SEISMED 1994). In addition, due to the enormous volume of data involved in multimedia data communications, there are also questions that must be addressed regarding the need for compression and how Message Authentication Codes (MACs) may then be used. In the longer term, the fact that uses of the TTP would not be restricted to the healthcare domain could aid its introduction and acceptance at the national and international levels. The use of TTPs in the healthcare context is described in more detail in (Purnell and Sanders 1995).

EXAMPLE SCENARIO

This section presents an example scenario to illustrate how future multimedia data exchange would be likely to function within and between HCEs. This is, in turn, used to highlight the need for security at the various stages involved. To this end, the information flows involved in a potential multimedia healthcare system are illustrated in figure 2 and explained in the description below.

The neurology department in one establishment (HCE A) performs a series of tests which produce a set of "raw" results data. However, HCE A lacks the equipment required to process and visualise the data, making it necessary to involve another site (HCE B). Once visualisation has been performed the results are transmitted back and stored in a database, from where they are subsequently accessed by a consultant at HCE A. However, further expert opinion is required and advice is, therefore, sought from another neurological consultant located at HCE C. Hence, the data is exchanged further, with the additional interpretation finally coming back to the originating consultant (allowing a more informed care decision to be made at HCE A). The consultants at HCEs A and C have access to a video conferencing link from their camera-equipped workstations, whilst the other parties in the scenario use standard workstations without such a facility.

From this basic outline, a general security specification can be given based upon the strategy described earlier. The different HCEs would communicate via local and national level TTPs, with all parties being authenticated by their local SMCs. Given that their workstations are equipped with cameras, the two consultants could potentially be authenticated by an image recognition system. However, the data production and data processing centres, utilising standard workstations, would have no facility for multimedia-enhanced authentication methods. Authentication of these parties would, therefore, be reliant upon the SMC facilities for activity supervision (possibly alongside traditional methods). The example is heavily communications oriented and the SMCs would communicate via the TTP hierarchy to authenticate and validate the various data exchanges and messages. The principal services required between HCEs A and B would be data integrity and confidentiality, whereas the HCE A / HCE C link would also require that the consultants were unable to repudiate information messages added to the system.

The example primarily illustrates the types of information exchange and consultations that the use of multimedia in healthcare will make possible. It also serves to underline the need for secure data communications between the various parties involved. The use of the TTP / SMC hierarchy would ensure that security was consistent across the three sites involved; a factor that considerably reduces the potential problems of sharing data and facilities as discussed.

CONCLUSION

The need for security is not unique to multimedia-based systems - indeed, similar demands already exist in many operational healthcare applications. However, the important point is that introduction of multimedia will serve to increase the demands significantly. Neither is
the proposed security strategy restricted to applications within healthcare establishments. However, the primary reliance upon software methods makes it particularly suited to HCEs, which are often more significantly financially constrained in relation to security than other types of organisation.

Our group is currently involved in the development, implementation and evaluation of a prototype multimedia patient records system in co-operation with a local HCE. Security is being considered as a key issue the project, with elements of the proposed strategy being addressed. It is hoped that the research will also help to identify other considerations that arise from the practical implementation of multimedia in healthcare.

The adoption and utilisation of multimedia technologies in healthcare is accelerating and it is likely that there will be a period of transition as research projects and pilot programmes (such as the EC 4th Framework) proceed in this area and produce their recommendations. From these, the principal uses and benefits of multimedia within healthcare will be established. We believe that it will be important for security issues to be considered during the planning and development of future systems, as the nature of the environment could well make it more difficult to securely integrate suitable protection later (or at least without it appearing to be an obvious afterthought).

REFERENCES


SECURITY MANAGEMENT IN THE HEALTHCARE ENVIRONMENT

S. M. Furnell and P. W. Sanders

Network Research Group, Faculty of Technology, University of Plymouth, United Kingdom.

ABSTRACT

Modern healthcare establishments are increasingly reliant upon information systems in all aspects of work and any compromise of their security may represent a significant threat to both the organisation and the patient. This paper discusses the increasing need for standardised levels of protection in healthcare computing systems and networks, outlining steps that have been taken to achieve this within European establishments. The paper then considers specific technical concepts that may be applied to improve security in healthcare at both local and international levels.

1. INTRODUCTION

As with many other areas of society, the healthcare field has been significantly affected by the adoption of information technology. Modern establishments now utilise a wide variety of equipment, ranging from standalone PCs to minicomputer or mainframe systems, representing significant assets of the business. In addition, many organisations now incorporate links to remote sites via Wide Area Network (WAN) arrangements, with increasing volumes of data transmitted between different establishments. This is likely to increase still further with the proposed standardisation of computerised health records using a common data structure [1].

The adoption of information technology has been accompanied by an increase in the number and variety of medical applications, which now affect most areas of operation (including patient care, finance, staffing, administration and many more). As a result, healthcare professionals have become increasingly dependant upon the availability of computer systems and reliant upon the correctness of the data that they hold.

The above trends highlight an increasing need for security in healthcare systems. Information systems may be compromised by a variety of accidental acts or by deliberate, malicious activity (e.g. hacking, fraud, virus infection and the like). As such it is now recognised that security issues must be considered during the design and development of new health information systems. In addition, security must also be added or enhanced in many existing systems, which were originally implemented without such considerations in mind and consequently have no standard arrangements.

2. SECURITY REQUIREMENTS IN HEALTH CARE

As with many other application areas, security requirements in healthcare are centred around the issues of confidentiality, integrity and availability [2]. These may be achieved by incorporating security services for authentication, confidentiality, integrity and non-repudiation as defined by ISO [3].

The nature of the healthcare environment tends to impose constraints on the types of protection that will be considered acceptable. For example, measures that greatly interfere with users abilities to perform their primary duties (e.g. care delivery) will not be tolerated. This points to a requirement for measures that are as simple and transparent as possible. In addition, financial cost is an important consideration as investment in security is often hard to justify against expenditure that would improve patient care. As a result, the use of
Software-based technologies may be a more realistic approach for widespread adoption than expensive hardware-oriented methods.

Despite these constraints, the increased interconnection and sharing of data between different establishments heightens the need for uniform levels of protection throughout the healthcare community.

3. BASELINE SECURITY FOR HEALTHCARE SYSTEMS

The need for improved security is already recognised within Europe and has been addressed by the CEC SEISMED (Secure Environment for Information Systems in MEDicine) project, with which our group has been involved [4]. The objective of SEISMED is to provide practical security advice and guidance to all members of the healthcare community who are involved with the development, operation and management of information systems.

Part of the project has been dedicated to the development of baseline security standards for existing systems and networks, describing the levels of protection that are considered appropriate for the healthcare environment. It is envisaged that these will eventually help to form a common reference for the security of healthcare systems within Europe.

The guidelines for existing systems highlight ten key principles of security which must be considered: (1) security policy and administration; (2) physical security; (3) disaster planning and recovery; (4) personnel security; (5) information technology facilities management; (6) user identification and authentication; (7) system access control; (8) database security; (9) system maintenance; (10) legislation compliance. These principles encompass a very wide range of considerations, with coverage ranging from general security concepts to more specific technical measures. In addition, the networking of medical systems has been recognised as an important issue in its own right. Whilst networks offer significant opportunities for improving healthcare services (thanks to the increased availability and sharing of information), there are also inherent security considerations. Examples of network threats include wiretapping, message replay, message repudiation and user impersonation. The SEISMED guidelines for networks present a further set of baseline standards to counter these and other threats and are primarily based upon encryption.

The definition of a health care baseline represents a significant step in achieving the desired standardisation of protection in the field. However, whilst the baseline standards provide comprehensive guidelines on "what" aspects of security should be considered, they do not attempt to describe in any great detail "how" technical measures may be best implemented. A comprehensive and flexible security system is needed that can be integrated into applications as required. The remaining sections outline how such a system may be realised in the health care environment.

4. USE OF TRUSTED THIRD PARTY TECHNIQUES

To meet the more specific network security requirements for both local and wide area systems a unique and unforgeable identification of all potential users (perhaps on a global scale) is necessary. These identities must be authenticated and "binded" to the activity or data used in that session. A naming and registration policy and infrastructure based on the international standards and technical framework of X509 / ISO 9545-8 [5] may be appropriate. Non-repudiation of the activities is required, together with confidentiality and data integrity during communication. Most methods to achieve these services are based on secret key cryptography and involve digital signatures, the encryption of data and the support of Trusted Third Party (TTP) infrastructures for wide scale use.

The implementation of such an arrangement involves public key systems, such as the RSA algorithm, with smart card technology for transparency and ease of use by the healthcare staff. The cards perform various cryptographic functions (the creation and verification of signatures, encryption / decryption of data, storage of secret keys and other sensitive data) and perform other special functions particular to the application. The TTPs act mainly as Certification Authorities for the digital signatures that they provide and, whilst they give a value-added service, must be trustworthy beyond the level of normal computer systems.
In order to provide all the necessary functions on an international scale a network of TTPs is required, as shown in figure 1. At this level the infrastructure will be generic for all applications, but at the local domain and sub-domain levels (as shown in figures 2 and 3) specific operations can be incorporated to satisfy HCE security policies, with the TTP being extended to a more comprehensive Security Management Centre (SMC) set of functionalities [6].

Figure 1. TTP infrastructure at an international level

Figure 2. TTP infrastructure at a national level

Figure 3. TTP infrastructure at a local level
In order to guarantee the authenticity of certificates a hierarchical certification structure is used. This is shown in figure 4, along with the format of the certificates, illustrating how additional certification occurs at each TTP level (with SK and PK representing the secret and public keys of the TTPs at each stage).

![Logical TTP hierarchy](image)

Each TTP in the hierarchy is certified by the TTP in the next layer up, which not only provides credibility of the complete system by defining the individual certification path within a certificate, but allows for the loss of a hierarchical level under fault conditions (with the next higher order certificate being used). The arrangement is common to the X509 Directory services architecture.

The TTP network can provide or verify signatures via the certificates, facilitating authentication and non-repudiation services. In addition, secret keys can be passed between users in a hybrid system where a symmetrical algorithm is used to provide confidentiality. Finally, integrity of data can be confirmed by the signing of a Message Authentication Code that is a hash function of the message.

As previously mentioned, additional security services can be incorporated into the TTP overlay in the local HCE security domains. This is discussed in the next section.

5. REAL-TIME SUPERVISION

Whilst the TTP will ensure the integrity and confidentiality of operations, an additional mechanism may be required within the local HCE domains to ensure that users are continually authenticated during their session and that they do not act outside their permitted bounds. A solution is to incorporate a real-time supervision system to detect unauthorised activity and strengthen standard authentication and access controls.

The supervisor would use expert system techniques to compare user and process activities within the domain against models of normal and suspicious behaviour, thus revealing any potential security problems (i.e. if an activity is incompatible with normal behaviour or is compatible with suspicious behaviour then it may be an intrusion). These models may be represented by maintaining behavioural profiles (for normal activity) and using pre-determined intrusion indicators (for suspicious activity).

It is considered that behaviour profiling may operate at 2 levels. At a high level it is possible to classify users according to their role within the HCE, developing general rules for acceptable activities within each
class. In addition, lower level profiles can be developed for individual users by analysing their use of the system. Measurable characteristics may include application and file usage, typical access times and locations, individual keystroke / typing patterns and instances of login failures or access violations. Validation of activity against the high level profile should ensure that users are operating within their legitimate bounds, whilst the lower level also allows authentication of the subject according to the behavioural characteristics. The user-specific profiles would need to be refined over time to account for legitimate changes in behaviour.

In addition to using profiles, the supervisor would monitor the system at a more general level to identify suspicious activities that may form part of compromise attempt. Examples of such indicators may include access of infrequently used files, consecutive access violations and extensive / frequent use of "help" systems. Whilst none of these events alone would be conclusive of an intrusion, they could be used as a trigger for more detailed monitoring or investigation. The disadvantage of this approach is that it will only cope with known intrusion scenarios.

Supervision could operate continuously throughout a session or at random periods, depending upon factors such as system load and application sensitivity. In either case, it would operate transparently unless an intrusion was suspected (in which case the system manager would be alerted and / or other appropriate safeguards would be taken).

The implementation of supervision in this manner is compatible with the desire for a software oriented approach to security as described in section 2.

6. CONCLUSION

A European-wide network is already operating on a prototype scale [7], with extensions to the HCE being designed at present. It is expected that this approach will provide a relatively cheap and simple to use service, facilitating effective security for health care establishments.

7. REFERENCES

THE USE OF SIMULATION IN COMPUTER-BASED SECURITY SYSTEMS

Steven M. Furnell, Peter W. Sanders and Colin T. Stockel
Network Research Group
University of Plymouth
Plymouth, United Kingdom
E-mail: stevef@soc.plym.ac.uk

Keywords: Computer systems, Statistical analysis, Real-time simulation.

ABSTRACT

The aim of this paper is to examine the applicability of simulation techniques to the testing and evaluation of computer security systems. Attention is specifically focused upon a relatively new area of security, namely advanced user authentication and supervision systems that are able to detect intrusions in real-time, based upon the comparison of user activities to predetermined behaviour profiles. The discussion is supported by the examination of a prototype monitoring system, based upon a simulation of the real-time analysis of user’s typing characteristics. The paper also considers a number of inherent problems in simulating the operation of a security system.

INTRODUCTION

Recent advances in the complexity of information systems, networking and telecommunications technologies have dictated an increasing requirement for more advanced security systems to safeguard against accidental and deliberate damage to systems and data.

Traditional approaches to user identity verification (principally passwords) can be considered increasingly inadequate as information system usage becomes an ever more routine part of society. The ability to utilise one system to access a multitude of others via global networks requires that user authentication be dependant upon more than just one (or a small series of) discrete judgement(s). In addition, it is desirable that mechanisms are incorporated that do not overburden the user with security responsibilities. However, even the more secure techniques available, such as the use of smart cards, require significant positive action on the part of users in order to be authenticated and may also be costly to implement on a large scale.

In light of such considerations it is increasingly desirable to redirect the focus of identity verification away from the user to being more of a system responsibility. An area of activity that supports this view is the development of advanced user authentication and supervision systems that aim to detect computer-based intrusions in real-time. These attempt to categorise various behavioural characteristics of legitimate users to form profiles of their normal system usage that can then be used as the basis for future identity verification and supervision (Lunt 1990; Bauer and Koblentz 1988).

However, such intrusion detection systems are, by definition, more complex than traditional means of authentication and access control and, as a result, the issue of effectively testing them may be considered problematic. Testing can no longer be regarded as being simply a question of determining whether a particular security measure can be easily broken or bypassed (Robertson 1992). It is also necessary to get a measure of effectiveness (i.e. how successfully can genuine intrusions be detected without mistakenly disrupting legitimate activity). In addition, testing cannot be effectively conducted by an individual or even a small team. The very nature of the concept requires that many genuine examples of user activity must be used as the basis for testing which, in turn, dictates that a reasonably large and diverse group of test subjects must be involved. However, it would be impractical (and probably undesirable) to introduce such a system into a "live" environment until it is known to work effectively, otherwise its presence could disrupt legitimate work (e.g. by causing the false rejection of valid users). There is also the consideration that the use (or simulation) of intrusion scenarios in an operational environment would
be a questionable proposition, as it could adversely affect system operation and potentially damage data.

SIMULATION IN SECURITY SYSTEM TESTING & EVALUATION

The considerations identified above highlight a significant requirement for off-line testing and evaluation of security systems, but in a context that will still provide a realistic measure of effectiveness.

An approach to the problem is to carry out the testing of such new security systems in a simulation environment, but using behavioural information taken from actual user sessions. In this way, data relating to user actions could be "recorded" from genuine sessions and then subsequently replayed, off-line, into the security system for analysis. This is illustrated in figure 1 below.

![Figure 1: Simulation in security system testing](image)

Keystroke analysis provides a behavioural biometric measurement based upon distinctive characteristics of the users typing styles. When used in the context of an intrusion monitoring application, the technique may be used to provide a continuous and transparent means of verifying user identities in conjunction with their normal system activity.

An experimental system, comprising three modules as listed below, was developed for the PC environment to allow an evaluation of the concept in practice:

- **Profiler**
  Accepts reference typing samples in order to create profiles for legitimate users.

- **Sampler**
  Accepts additional user typing samples and stores all keystrokes and associated timings for later use.

- **Monitor**
  Compares the typing samples against profiles to determine the effectiveness of the system, simulating the real-time entry of the sampled keystrokes.

These elements fit broadly into the structure that was presented in figure 1, with sample entry equating to the on-line activity and the monitor module representing the off-line security system simulation.
Tests were conducted involving 26 typists, with typing profiles being created based upon the average inter-keystroke times exhibited when entering specific character pairs (digraphs), with mean and standard deviation values being maintained for each pair. Subsequent supervision attempted to verify user identity by comparing incoming keystrokes against the relevant profiled values, with incompatible times being judged as invalid. If either the overall percentage or number of consecutive invalid keystrokes exceeded certain user-specific thresholds an impostor alert was raised.

The aim of the investigation was to establish the impostor false acceptance rate (FAR) with a false rejection rate (FRR) for valid users of 0%.

Uses of simulation elements

Simulation aspects were incorporated into the study in a number of ways:

- the use of stored user typing samples (including inter-keystroke timing data) to simulate the entry and analysis of keystrokes in real-time;

- the simulation of intrusion scenarios by using "non profile owner" typing samples as impostor inputs to the system. This allowed approximately 1300 impostor test cases to be derived from just 26 test subjects;

A further potential use of simulation that was identified (although not extensively explored) was the ability to generate simulated impostor typing samples based upon data from the initial user profiles. This process would work as follows. After selecting some text as the basis for the test sample, the character digraphs within it could be extracted and matched against the associated mean and standard deviation values held in the profile chosen to represent the "impostor". Using the upper and lower limits of the standard deviation from the mean to define a valid range, a random value could then be generated to represent the impostor's inter-keystroke time for that digraph. For example, if the text contained the digraph "TH" and this had been profiled with mean of 121ms and standard deviation of 47ms, a valid range would be defined as below:

  74ms <-------- TH ---------> 168ms

So a typical inter-keystroke time generated by this impostor might be 106ms. This process would continue throughout the entire text to create an appropriate test sample simulation.

To make the sample even more realistic, the simulation could also take into account the maximum percentage of invalid keystrokes that the "impostor" would generate against his / her own profile (given that profiles also maintain this value, for use as an authentication threshold). To this end, a further random element could be introduced by generating an appropriate proportion of keystroke times in the test sample incompatible with the host profile.

Once generated, these simulated test samples could be used as a realistic means of testing the false acceptance rates against other user profiles (thus allowing a much more comprehensive test of the system's impostor detection effectiveness without requiring any further test subject involvement). It should be noted, of course, that these artificial samples would only be useful as "impostor" attempts. They could not be used to test legitimate user performance as the creation process would always ensure that they were compatible with the host profile.

The desirability of using such artificially created samples has previously been identified by other research in this area (Brown and Rogers 1993), but in the more limited context of user name entry. The technique was not used to contribute to the results from our study that are described below, but would have been particularly useful had insufficient genuine test subjects been available to participate.

Results and discussion

The results obtained provided a number of useful indications regarding the effectiveness of keystroke analysis as an intrusion detection technique. From the 1300 impostor cases that were used, a FAR of 15% was observed (which can be regarded as a "worst case" figure given that no false rejections occurred). Although this value is somewhat high, it should be remembered that keystroke analysis could be implemented as just one aspect of a more comprehensive intrusion monitoring system and, therefore, other factors could also be introduced that would compensate for the currently undetected cases. Of the detected cases, 49% were due to the percentage of invalid keystrokes observed, whilst the other 51% were due to consecutive invalid keystrokes entered by the impostors.
Given that impostor detection was achieved in the majority of cases, the other important consideration was how quickly it occurred (i.e. how many keystrokes would the impostor have been able to enter before detection). This issue was also addressed by the study, and the results observed are shown in figure 2. This indicates the proportion of detections that occurred within each of five distinct keystroke ranges (based around 40 character blocks - equivalent to half a standard line of text).

Finally, the establishment of a “worst case” FAR rating would not have been possible outside of a simulation environment. The ability to specifically configure the system allowed an FRR of 0% to be ensured, with successful impostor performance then being observed at this level. Conversely, the simulation could have been used to determine the level of false rejections with a guaranteed FAR of 0% (however, this approach was not pursued as rejection of legitimate users would be extremely undesirable in the context of a continuous monitoring system).

A more detailed description of this study, the methods involved and the results obtained can be found in (Furnell 1995).

**POTENTIAL PROBLEMS OF SECURITY SYSTEM SIMULATION**

Even with simulated environments and intrusions there are still a number of considerations that complicate the issue of security testing. A principal point here is that many successful intrusions / system security breaches result from scenarios that were either unanticipated or overlooked by system designers. This is evidenced by the details of known abuse cases (Audit Commission 1990) and also by the fact that, despite the many controls that are present in existing systems, around half of the detected cases of computer abuse are only discovered by chance (Audit Commission 1994).

In addition, there may be difficulties associated with simulating the security environments. Keystroke analysis is quite a trivial example in this respect, whereas most other potential candidates for behaviour profiling (e.g. usage of operating system commands and applications) would require more complex simulation environments and would also demand that profiles were developed over a longer period than in the study described.

Finally, there are a number of important aspects that the approach (as discussed) cannot address. These include issues such as the systems compatibility with other applications, processing overheads that may be incurred in a live environment and acceptability to end-users. As a result, there is still a need for system evaluation in the context of a live “pilot” study, but with the major question of effectiveness having largely been answered.

Fig. 2: Keystrokes before impostor detection

It can be seen from this that the vast majority of impostors would be detected within 160 keystrokes. This result was also considered reasonably encouraging, although it is acknowledged that if intruders were of a particularly malicious nature, then they would possibly require somewhat less than 160 keystrokes in order to cause significant damage.

The experimental study itself would have been considerably more difficult to conduct had the simulation element not been involved. With the test samples being obtained and stored, the study only required that subjects were available for a maximum of around an hour (much less with the faster typists). If the testing had had to be conducted in real-time, on an individual basis, it would have entailed repeated test sample entry and increased the subject availability requirements to such an extent as to make a large test group impractical. In addition, the prototype authentication system would need to have been installed on individual user systems - potentially disrupting their normal activities.

A further point is that the simulation-based environment provided the ability to re-mount the experiment after re-configuration of various aspects of the system (e.g. user authentication thresholds, the number of recent keystrokes monitored, valid inter-keystroke ranges). This facility was used to allow the optimum monitoring configuration to be established.
CONCLUSIONS

The provision of effective security still remains a key issue in the implementation of information systems. Whilst information technology has already affected most aspects of society (e.g. government, healthcare, policing and commerce), this has largely occurred in the context of "closed" systems. As formerly independent domains merge and share common global networks, the requirement for adequate security will increase still further.

It is hoped that this paper has served to highlight how simulation techniques may have a useful role to play in the security field, targeting an approach to protection that is considered appropriate to the perceived needs of future systems.

Our own study served to demonstrate various areas in which simulation could be involved in a practical context and proved how it could vastly improve the ease of testing in this type of system (with the results of the evaluation indicating the significant potential of keystroke analysis as an intrusion monitoring / user supervision technique).

As information systems advance, it is envisaged that intrusion monitoring systems at this level and beyond will become increasingly more attractive. As such, the use of simulation approaches similar to that discussed will be ever more appropriate.

REFERENCES


BIOGRAPHY

Steven Furnell graduated from the School of Computing, University of Plymouth with a first class honours degree in Computing & Informatics in July 1992. Since August of that year he has been a post-graduate research student in the University's Network Research Group undertaking a PhD programme. The project is entitled "Data Security in European Healthcare Information Systems" and, in addition to addressing intrusion monitoring issues, has also involved the development of security guidelines for European healthcare establishments. The research is being supervised by Peter Sanders and Colin Stockel.
SIMULATION OF A MULTIMEDIA PATIENT RECORDS SYSTEM

Nichola J. Salmons, Steven M. Furnell, Peter W. Sanders and Colin T. Stockel
Network Research Group
University of Plymouth
Plymouth, United Kingdom
E-mail: nichola@soc.plym.ac.uk

Keywords: Health care, Information systems, Man-machine interfaces.

ABSTRACT

The paper highlights the need for, and the benefits of using, simulation during the development and implementation of modern healthcare systems. Whilst healthcare establishments already utilise information systems in a wide variety of disciplines, the majority of systems are currently isolated, with patient records largely based upon manual methods. As such, it is envisaged that the establishment of composite, multimedia-based patient records would considerably aid care delivery. After a brief discussion of the advantages that this would bring, the paper proceeds to highlight how simulation can be employed to aid system design and development in a number of areas (including the user interface, records structure, security, networking requirements and the profiling of future application demands). The discussion is based upon work currently being conducted by the authors within a practical research project.

INTRODUCTION

During the past twenty or more years computerised information systems have gradually been introduced to, and utilised within, a large number of Health Care Establishments (HCEs). Modern medical care requires the use of computerised systems to process, visualise and store vast amounts of information. The data produced by these more advanced medical systems consists of not only simple textual data but also digital images, full motion video, audio and visualised graphics (Nelson and Todd Elvins 1993). The use of computerised systems, both centralised and departmental, has resulted in HCEs being able to offer ever more complex and comprehensive medical care.

INFORMATION TECHNOLOGY IN MODERN HEALTHCARE

Computers now form an integral part of the process of administering and monitoring patient care. Additionally, computerised systems have also enabled a wide range of complex scanning and diagnostic procedures such as Computer Tomography (CT), Magnetic Resonance Image (MRI) and Ultrasonic Imaging to be offered (with the information gained then being utilised in the planning and delivery of further medical procedures). The increased use of information technology has resulted in clinicians being able to collect, generate, analyse and interpret ever greater amounts of patient data. The availability and quality of this data then enables the clinicians to prescribe and administer the most appropriate healthcare programme for the patient.

However, at present, within many HCEs there are growing problems associated with the management and organisation of the rapidly proliferating amounts of both patient data and management/administrative information. Due to the fragmented development and implementation of the HCE information systems, there tends to be little or no integration or exchange of data between systems. The lack of information organisation, in conjunction with the sheer volume of data, can often result in decreased clinical efficiency, as more time is spent attempting to search for and retrieve data from different systems. Thus the benefits offered by the availability of increasingly comprehensive patient data are diminished and, therefore, in order to improve the situation data needs to be made more portable, accessible, comprehensible, and appropriately structured.
It is widely envisaged that these problems could be overcome by the adoption of composite patient healthcare records, based around multimedia technology (Treves et al. 1992).

ADVANTAGES OF A MULTIMEDIA-BASED HEALTHCARE RECORD

The use of multimedia patient data in healthcare has already begun and will inevitably increase as more clinicians are afforded the opportunity to produce and utilise high quality data at a relatively low cost. There are currently two developmental paths to the production and utilisation of multimedia data within healthcare. The first is that offered by the ability to obtain “raw” data via advanced techniques such as MRI and CT, which can then be visualised, manipulated, rendered and animated by powerful workstations, to generate the desired end result. At present this route is expensive due to the data collection and manipulation tools required, although it is already implemented in larger HCEs. The other path is that offered by the PC, where technological advancement is now reaching the point where clinicians can produce high quality multimedia data (including video, audio, graphics, images and text) both easily and relatively inexpensively.

Thus the way is clear for clinicians to be able to create and utilise multimedia clinical data. A composite multimedia record would improve the provision of patient care, as clinical decisions would be made with all the multimedia patient data available on one system, in the most easily comprehensible and informative manner.

The ability to view patients records easily will in turn enable clinicians to more comprehensively assess patient needs, responses to treatments, and on-going progress and may aid clinical decision making. Thus the patient will benefit from the use of multimedia data, in that they will be prescribed the most appropriate care plans. In addition, the healthcare providers benefit from the comparative cost reductions facilitated by the administering of the most suitable patient care.

The proposed system would ideally be able to integrate with any existing systems holding patient data, meet the desired user requirements, be secure against malicious or accidental intrusion, facilitate data communications within and between HCEs and be able to accommodate future medical advances and changes in working practices (Orozco-barbosa et al. 1992).

However, the introduction and implementation of multimedia patient records may prove to be problematic if there are not accompanying advances and improvements in the structuring, integration, portability, accessibility and comprehensibility of the data generated.

DEVELOPMENT OF A SIMULATION-BASED PROTOTYPE SYSTEM

This section examines practical work that is being undertaken by the research team to help realise the composite multimedia healthcare record concept. The general background is discussed, followed by a description of the simulation aspects involved.

Project Aims and Background

The remit of the project was to establish where the use of multimedia would be most applicable in healthcare and to define the structure, content and interfaces required for a multimedia-based records system. Additional considerations were the definition of most effective systems working practices, with the procedures required for the creation, appending, manipulation and management of the patient data.

The systems development was based at Derriford Hospital, a major HCE local to the research team. In terms of information systems, this establishment is similar to numerous others in the UK. Apart from a centralised Patient Administration System (PAS - which is accessible from all departments), a few independent departmental systems and a number of specialised stand-alone machines, the majority of patient data is generated and maintained manually.

It was established through interviews (described below) that the use of computers is alien to the majority of hospital personnel, with a worrying (and widely held) perception that computers will not form part of the future for healthcare. This view was generally based upon the belief that the computerisation of many operations and working practices would be costly and offer no real advantages. These factors suggested that the development of a simulation-based prototype would be the best way for the project to proceed, as this would allow an opportunity to demonstrate the future possibilities and benefits that would be offered, breaking down the resistance of the users.
Research Methodology

The task of developing a composite, multimedia records system is obviously immense. For this reason the scope of the study was limited, with a single department being selected to act as the "base" for the project. It was considered that the base should be a department in which there would be a number of opportunities for the introduction and use of multimedia patient data and one in which the patients are often referred to and between a number of closely associated departments over long treatment periods. As such, the Ear Nose and Throat (ENT) department was selected, with Radiology, Speech Therapy, Plastics, Microbiology, Dental Specialities and Maxillo-Facial departments as peripheral or closely associated referral departments.

The research method selected was that of performing discursive interviews throughout the selected departments. A range of staff were covered, from consultants to secretaries, so that the full scope of the departmental operations could be assessed. The data obtained was then used to create a prototype system which would then undergo recursive refinements. The desired system requirements and established working practices, along with user and departmental data exchanges and paths, were then abstracted and modelled from the interview results.

A significant issue in the design of the system was ensuring integration with current, and possible future, clinical practices. To this end, clinical staff were asked to identify "core non-flexible" and "core flexible" clinical and administrative practices and procedures. The "core non-flexible" practices and procedures were those which it would be impractical to change to any extent and which must, therefore, be maintained whether the patient records system was computerised or not. The "flexible" practices were those which could be re-engineered so long as the desired end result was still achieved.

The "non-flexible" practices tended to be made so by being either time sensitive (e.g. the requirement for immediate clinical reporting of results within the Radiology department, as delays could potentially compromise patient health) or a matter of established medical convention or clinical practice (e.g. that departmental appointments are always made internally). As such, the departments involved would find it impractical to perform them in any other way.

The "flexible" practices were those which could be made easier by the computerisation of the Patient Records System. These included the ordering and tracing of patient notes, the appending of data, and the searching for clinical details.

The interviews also established where it would be clinically appropriate to generate the multimedia data which would be used within the proposed records system. The selected departments each considered where, within their clinical discipline, it would desirable to obtain multimedia patient data (for instance, when would it be desirable to have video data of the patient, and what were the practicalities of generating it?).

Uses of Simulation

Having used the interviews to establish the basic system requirements, the study could proceed to consider prototype implementation.

It is envisaged that once an initial prototype is developed and in-place at the hospital, simulation will form the core of its future development. A cross section of users will initially simulate the typical everyday use of a small number of demonstration multimedia patient records. From this the desired systems interface can be established. A number of different records structure styles can be offered, with the users then determining which is easiest to use. Different clinical scenarios will be simulated, which will require the records to be manipulated in a number of different ways.

From the record usage simulations a comprehensive range of individual record search options will be defined. These will indicate and define those data items and criteria (such as previous surgery, previous treatments, current and past medication, family history, noted medical conditions, etc.) by which the records need to be searched.

Once the use of individual records has been simulated, the project will move on to simulate a system dealing with a number of records, defining the functionality required with respect to multiple records. The users will be able to define the searches, and other functions, which the system must be able to perform between separate multimedia patient records. Thus at the end of the record usage simulation stage the preferred user interface, record structure, and intra and inter-record functionality will have been defined.
The next stage will be to simulate record creation and maintenance. Simulation in a real clinical environment will enable the clinicians to determine where, and when, it is practical to obtain multimedia patient data. The data collection processes must not intrude upon, or compromise, clinical working practices. The staff must then simulate the editing of the patient data, and the record appending practices required, again in a manner which integrates with other working practices.

Simulation will, therefore, enable the users to define and develop the most suitable practices for the collection, processing and maintenance of the multimedia patient record data. If these procedures can be made as simple and easy as possible then users, both clinical and administrative, will be far more inclined to pursue the use of multimedia in healthcare. The simulation environment may then be extended beyond this to consider other important aspects relating to multimedia records system implementation, including security, network requirements and additional functionality.

The requirements for data security can be considered and various approaches simulated. It is envisaged that the multimedia context will require an approach to security that is as transparent as possible, so as not to unnecessarily detract from the otherwise user-friendly nature of the environment (Furnell et al. 1995).

As an example, whilst user authentication could principally be based around a traditional password approach, it might be desirable to evaluate more friendly (and secure) methods within the context of the simulation environment. Alternatives could include the use of smart cards, real-time supervision systems (verifying identity by analysing factors of user behaviour such as typing styles and application usage) and/or various biometric identification techniques that might be feasibly implemented using existing multimedia hardware (e.g. faceprint or voice recognition). Through the simulation study, appropriate techniques or combinations could be established as required by different user groups.

There will also be a need for security restrictions at the departmental and user levels to control access, modification and deletion of different aspects of the overall records.

Once the security aspect has been simulated, the study can move on to define aspects of the systems network requirements and possible additional functionality. The users will continue to simulate the everyday use of the system, but it will be extended to include additional features. These will include a range of departmental administrative, clinical audit, and management functions. A number of the proposed functions will reference the patient records data, whilst others will reference other data sources, some localised and some remote.

At this point the prototype simulation will not only be defining the desired additional system functionality, but will be helping to determine the systems integration and network requirements. By simulating the additional system functionality, the simulation will be able to establish those existing, or proposed, hospital systems from which data will need to be accessed. Thus the system integration requirements will be defined. The systems networking requirements will also be eluded to by the simulation of the additional functionality. From the use of the prototype it will be possible to determine the quantities of non-localised data required by the users, the data types required over the networks, and the acceptable system data throughput and response times, as well as the types and quantities of data transmitted by the users. Hence the simulation will give an indication of the systems network requirements. Security of data communications could also be considered here, with the simulation study considering various techniques that may be appropriate to ensure the confidentiality and integrity of transmitted data, as well as requirements for non-repudiation services (AIM SEISMED 1994).

The system simulation will also provide a valuable insight into the systems usage patterns with respect to the user types and help to determine the optimum working practices and duty ranges, for the different user types within the base department. Different systems operational modes may be simulated, in which the different user groups have subtly varying roles and duty ranges. From the simulation results the departments optimal operational mode can be established, thereby maximising systems and departmental efficiency.

CONCLUSIONS

At the end of the project, after the use of simulation as a development technique, in conjunction with the progressive implementation, extension and refinement of the prototype system, the users will obtain not a fully defined system but one which at least starts to address and overcome the numerous problems associated with
the development of a multimedia healthcare records system.

A simulation study as described would allow time for end users to become more familiar with the technology involved and would hopefully result in the development of a system which is of real benefit to them. The experimental period would also enable the clinicians to determine where, and to what extent, the use of multimedia patient data is most advantageous. The in-place use of a simulated prototype appears to be the only real option for the development of suitable systems as it is only through such an approach that the desired end result will be achieved.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. P. N. Gaunt, Mr. P. Windle-Taylor and Mr. M. Teague of Derriford Hospital for their generous assistance with the project. In addition, the members of the following departments are thanked for their time and co-operation during the interviews:

- Dental Specialties Department;
- Ear, Nose & Throat Department;
- Maxillo-Facial Department;
- Plastic Surgery Department;
- Radiology Department;
- Radiotherapy Department.

REFERENCES


BIOGRAPHY

Nichola Salmons graduated in 1991 with a BSc. in Applied Chemistry from the then Polytechnic South West (now University of Plymouth). This was followed by a PgD. in Computer Technology and Software Applications from the University of Central Lancashire in 1992, and an MSc. in Telecommunications Technology from Aston University in 1993. She is currently working towards her PhD at the University of Plymouth, the aim of which is to develop a "Composite Multimedia Healthcare Record". Steven Furnell is currently collaborating in this work, which is being supervised by Peter Sanders and Colin Stockel.
Development of
Security Guidelines for Existing Healthcare Systems

S M Furnell, P W Sanders and M J Warren

Network Research Group, Faculty of Technology, University of Plymouth,
Plymouth, United Kingdom.
E-mail: stevef@soc.plym.ac.uk

Abstract

As modern healthcare establishments become increasingly dependent upon information systems it is vital to ensure that adequate security is present to safeguard the confidentiality and integrity of data and the availability of systems. Whilst this is now generally recognised in the design of new systems, many existing operational systems have been implemented without security in mind. This paper describes the need for a standardised approach in the protection of existing healthcare systems within Europe and presents an overview of a new set of information security guidelines that have been developed specifically for the medical community.

The guidelines discussed have been produced as a deliverable of the Commission of European Communities (CEC) SEISMED (Secure Environment for Information Systems in Medicine) project, under the Advanced Informatics in Medicine (AIM) programme.

1 Introduction

The increasing accessibility of information technology (IT) systems during recent years has had a significant effect upon the healthcare field. Many healthcare establishments (HCEs) now operate heterogeneous IT environments with equipment ranging from standalone PCs to minicomputer and mainframe installations.

The influence of information systems can now be seen in most areas of healthcare operation, with an ever increasing number and variety of medical applications. In addition, IT also facilitates the exchange of medical data between different HCEs at both national and international levels. A significant result of these advances is that healthcare professionals have become increasingly dependant upon the availability of systems and reliant upon the correctness of the data that they hold.

As the adoption of information technology has increased so too has the requirement to protect the systems and the information they store. Healthcare systems may be vulnerable to a variety of accidental or deliberate threats and, as such, it is now recognised that security issues must be considered during the development and implementation of new health information systems to maintain the confidentiality, integrity and availability of the data held. Unfortunately, a significant proportion of
operational healthcare systems were originally designed and implemented with inadequate security and, as a result, security must also be added or enhanced in many existing systems.

2 The AIM SEISMED Project

The issue of information security in healthcare has been addressed by the CEC SEISMED (Secure Environment for Information Systems in Medicine) project, part of the Advanced Informatics in Medicine (AIM) programme [1].

The objective of SEISMED is to provide practical security advice and guidance to all members of the healthcare community who are involved in the management, development, operation or maintenance of information systems. The eventual aim is to establish a consistent framework for the protection of medical data across the European Union.

The project commenced at the beginning of 1992 with an original duration of 3 years, but this was subsequently extended for a further 6 months (until mid-1995). A total of 14 workpackages were established, each addressing a separate aspect of healthcare security. Five European HCEs (located in the UK, the Netherlands, Switzerland and the Czech Republic) were selected to act as Reference Centres for the project, commenting upon and ensuring the viability of the recommendations made.

The problem of securing existing systems was addressed by workpackage SP07, the scope of which was to produce a comprehensive set of recommendations for the addition (or enhancement) of security in operational healthcare systems and environments. The principal objectives of this workpackage were:

- to produce guidelines on the level of protection that should be attached to existing operational healthcare systems;
- to provide guidance as to how this level of security may be achieved;
- to revise the approach based upon Reference Centre feedback.

Whilst various guidelines and standards for IT security have previously been developed, none have specifically targeted the needs of the medical community at a European level. The new guidelines are intended to provide a common source of reference for European healthcare establishments and are relevant to (and will affect) all categories of personnel.

3 Baseline Security Recommendations for Healthcare Establishments

In order to assess current security practice and attitudes within European establishments a survey was distributed to HCEs in 11 community countries [2]. Amongst other things, this allowed a broad assessment of existing systems to be made and revealed a significant variety in both the types of system in use (i.e. hardware,
operating systems and applications) and the levels of security provided. For example, whilst virtually all systems included some form of user authentication mechanism (even if only a simple password in some cases), the attention given to other aspects of security (e.g. disaster recovery, physical protection and auditing) was, in general, significantly less. Furthermore, the variety of techniques used to address a single aspect of protection indicated anything but a standardised approach (e.g. the types of authentication mechanisms variously utilised include individual passwords, shared passwords, challenge-response mechanisms and other methods - with likely inconsistency between similar systems).

It was considered that, in many cases, the disparity indicated by the survey had resulted from the lack of appropriate standards and guidance, with HCEs being unclear over both general security issues and the level they should aim for. The most appropriate strategy for improving the situation was, therefore, considered to be the definition of baseline recommendations for security, to provide a common foundation for all HCEs.

This immediately raises the question of what level of security should be specified. The nature of the healthcare environment, with the inherent requirements to maintain patient safety and confidentiality, demands that protection should generally be higher than in many other domains. As a result, the security requirements extend beyond the levels proposed by many existing standards.

The new baseline recommendations have been developed to satisfy the following aims:

- to represent a minimum acceptable standard for the security of operational healthcare systems and their associated environments;
- to be usable by all HCEs and staff within Europe;
- to allow a straightforward means of validating existing systems security to ensure compliance.

The development of the resulting guidelines was based upon an interactive approach, in close co-operation with the SEISMED Reference Centres and in consultation with other independent healthcare professionals.

From the outset it was established that the recommendations should address more than just the host system in isolation. Indeed, to provide comprehensive protection, several aspects of security must be considered:

- logical / system-based controls;
- physical and environmental protection;
- personnel procedures;
- policy and administration issues.

On the basis of these high level requirements, existing IT security guidelines and standards [3,4,5] were used in conjunction with suggestions from within the project to formulate initial recommendations. These were progressively refined and enhanced over time on the basis of Reference Centre feedback and comments from independent
healthcare personnel. This procedure provided the principal criteria for retention, addition or removal of guideline recommendations.

4 An Overview of Existing Systems Guidelines

The final *Security Guidelines for Existing Healthcare Systems* [6] are grouped under 10 key *principles* of protection, representing the main elements governing the security of existing healthcare information systems (having been agreed in detail with the Reference Centres). The principles are denoted by ESP followed by a unique reference code, as listed in table 1 below.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESP0100</td>
<td>Security Policy &amp; Administration</td>
</tr>
<tr>
<td>ESP0200</td>
<td>Physical &amp; Environmental Security</td>
</tr>
<tr>
<td>ESP0300</td>
<td>Disaster Planning &amp; Recovery</td>
</tr>
<tr>
<td>ESP0400</td>
<td>Personnel Security</td>
</tr>
<tr>
<td>ESP0500</td>
<td>Training and Awareness</td>
</tr>
<tr>
<td>ESP0600</td>
<td>Information Technology Facilities Management</td>
</tr>
<tr>
<td>ESP0700</td>
<td>Authentication &amp; Access Control</td>
</tr>
<tr>
<td>ESP0800</td>
<td>Database Security</td>
</tr>
<tr>
<td>ESP0900</td>
<td>System Maintenance</td>
</tr>
<tr>
<td>ESP1000</td>
<td>Legislation Compliance</td>
</tr>
</tbody>
</table>

Table 1: Existing Systems Security Principles

Each of the principles has a number of associated *guidelines*. These represent the specific security concepts or countermeasures that should be considered by the HCE to meet the requirements of a given principle. As established earlier, the consideration of existing systems encompasses a very broad range of issues and the overall coverage consequently extends from general concepts to specific technical measures.

The 10 protection principles are described in more detail below. In each case the general purpose of the principle is stated, along with a list of the main issues that are covered by the underlying guidelines (the overall number of guidelines pertaining to each principle is given alongside its title).

1. Security Policy & Administration (*5 guidelines*)

   *General Principle*
   
   A formal policy will provide clear direction and support for security within the HCE. Policy is formulated from the senior managerial level, with subsequent guidance provided to all levels of staff. Correct administration of and adherence to the policy should ensure the effectiveness of HCE security controls.

   *Main issues*:
• the need for a security policy;
• policy awareness issues;
• co-ordination and administration of security;
• use of specialist security personnel.

2. Physical & Environmental Security (22 guidelines)

General Principle
The generally open nature of HCEs and their high degree of public access dictates that physical security measures are a vital first stage of protection to prevent unauthorised access to computing equipment and facilities. Systems must also be safeguarded against a variety of environmental hazards that may adversely affect operation.

Main issues:
• physical access control;
• security of HCE equipment;
• protection against natural disasters;
• environmental controls;
• various procedural measures.

3. Disaster Planning & Recovery (7 guidelines)

General Principle
The continuous availability of Information Systems is essential to the operation of a modern HCE. It is essential that adequate plans are made to ensure the level of availability needed by the HCE can be maintained in the event of any catastrophe. Recovery of IT systems should be a component of an overall HCE disaster / recovery plan.

Main issues:
• continuity planning (development, testing and update);
• fallback arrangements;
• post-disaster procedures and controls.

4. Personnel Security (8 guidelines)

General Principle
The major security weakness of many systems is not the technology but the people involved. Many organisations are extremely vulnerable to threats from their own staff and, as a result, even the most comprehensive technical controls will not guarantee absolute security. There are, however, a number of personnel-related measures that can be introduced to help reduce the risks.

Main issues:
5. Training & Awareness (6 guidelines)

*General Principle*
Information systems security can only be maintained if all personnel involved in their use know, understand and accept the necessary precautions. Many breaches are the result of incorrect behaviour by general staff who are unaware of security basics. The provision of security training and awareness will make it possible for staff to consider the security implications of their actions and avoid creating unnecessary risks.

*Main issues:*
- the need for general security awareness;
- specific areas that must be addressed (job training, use of information systems);
- recommendations for internal / HCE training and awareness initiatives;
- use of specialist training courses;
- assignment of responsibilities for training.

6. Information Technology Facilities Management (31 guidelines)

*General Principle*
A variety of activities can be identified that are related to the normal day-to-day use and administration of information systems. All categories of HCE personnel (management, technical and general users) have responsibilities that must be addressed in order to maintain security in this area.

*Main issues:*
- system planning and control;
- the importance of maintaining back-ups;
- media controls;
- auditing and system monitoring;
- virus controls;
- documentation issues.

7. Authentication & Access Control (28 guidelines)

*General Principle*
It is essential that IT systems are protected by comprehensive logical access controls. Access should be guaranteed for legitimate users and denied to all others. *All* classes of user must be identified and authenticated before *any* access is
granted and further mechanisms must control subsequent reading, writing, modification and deletion of applications and data. There should be no method for bypassing any authentication or access controls. HCE users are unlikely to be satisfied with controls that intrude upon working practices and chosen schemes should be transparent and convenient in order to gain acceptance.

**Main issues:**
- requirements for user identification and authentication;
- password issues;
- system and object access restrictions;
- methods of control;
- access in special cases (e.g. system management, third parties, temporary staff).

8. Database Security (*21 guidelines*)

*General Principle*
Database security is concerned with the enforcement of the security policy concerning the disclosure, modification or destruction of a database system's data. Databases are fast becoming very important for HCEs. Over 90% of today's IT systems contain some kind of database and the value of information stored is now widely recognised as a major asset, far more important than any other software. However, databases also introduce additional security concerns (e.g. granularity, inference, aggregation, filtering, journaling etc.) and therefore warrant specific consideration.

**Main issues:**
- control of medical database software;
- organisation and administration of HCE database systems;
- database operation issues.

9. System Maintenance (*5 guidelines*)

*General Principle*
System maintenance activities merit special consideration given the opportunities that exist to affect the operation of the system. Unauthorised or uncontrolled changes to any aspect of an operational system could potentially compromise security and, in some cases, endanger life. Maintenance must therefore be carried out in accordance with well-defined procedures.

**Main issues:**
- controls to prevent unauthorised changes to and upgrades of HCE software, vendor software and operating systems;
- requirements for testing and acceptance.

10. Legislation Compliance (*5 guidelines*)
General Principle
Specific levels of protection may be demanded in order to comply with national and European legislative requirements, as well to satisfy internal HCE policy. Whilst the guidelines highlight the most basic requirements, this principle represents an ongoing process which must take account of any new legislation that may be relevant, as well as ensuring compliance with existing standards.

Main issues:
- data protection;
- abuse of information systems;
- prohibition of "pirated" software;
- compliance with internal security standards;
- retention and protection of business records.

5 HCE Target Audiences

It should be evident that many of the issues covered are not relevant to all HCE staff. As such, the Security Guidelines for Existing Healthcare Systems are targeted at three main staff groups (as shown in figure 1), with separate guideline sets having been developed for each audience.

Security Guidelines

- General HCE Staff (50 Guidelines)
- HCE Management (61 Guidelines)
- IT & Security Personnel (122 Guidelines)

Fig. 1: HCE target audiences

Whilst all three sets draw upon the same core principles, they nevertheless differ dramatically in terms of the type and quantity of information presented. The anticipated readership and general content of each set is as follows:

- The General guideline set is aimed at the majority of HCE staff, including clinicians, administrators and general system users. Guidelines are presented for user reference during day-to-day use of HCE information systems, highlighting what they can do to safeguard security.
The Management set primarily targets the senior decision makers within the HCE, who will be responsible for defining security policy (although a significant number of points will also be relevant at department / line management level). This set is intended to highlight areas in which management should be directly involved and also improve management security awareness by explaining / justifying the importance of other more technical guidelines (for which management approval will be required).

The IT & Security Personnel set is aimed at IT staff, system administrators, security officers and other support staff who will be most likely to have the lower level responsibilities for implementing security. This is the most detailed of the subsets and should be a key source of reference for implementation and validation of security.

The Management and IT & Security audiences would also be expected to read and observe the General guideline set.

6 Implementing the recommendations

The Security Guidelines for Existing Healthcare Systems should be applied in any European Healthcare Establishment with existing operational information systems (where the term Healthcare Establishment refers to any establishment providing medical services, research, training or health education). They will be relevant even where systems are thought to include security provision, so that the level of protection can be validated against the recommendations.

However, given the diverse nature of European healthcare environments and systems, it is impossible to specify precise guidelines for implementation. Establishments will differ in terms of both the information systems used, as well as financial, operational and other constraints that may apply. These issues will all have bearing on the applicability of the recommendations and the guidelines therefore concentrate more on describing what aspects of security should be considered rather than how they may be best implemented (with broad recommendations that should be compatible, to at least some degree, with the majority of systems and environments)

Despite these attempts to ensure applicability, it is still conceivable that some guidelines may not be suitable for all systems. As such, implementors must use their discretion in cases where guidelines are genuinely inappropriate to the environment. However, recommendations should be followed as closely as possible and in some cases the implementation of a guideline will depend upon others already being in place (which is made clear from the guideline context and / or cross-references to other points).

As for the implementation strategy itself, it would obviously be impractical to attempt to address all of the suggestions at once due to constraints of cost and likely disruption to services. A phased approach is, therefore, advised in which each principle is
considered in turn to identify the areas in which the HCE / department is currently deficient. The individual guidelines may then be assessed to determine implementation priorities based upon local requirements.

Further work within the SEISMED project has resulted in the development of the methodology SIM-ETHICS (Security Implementation Methodology - Effective Technical and Human Implementation of Computer based Systems) which may be used to assist with the implementation of these and other SEISMED guidelines [7]. The methodology is based upon the concept of participational management, using groups of users and managers to carry out a hypothetical implementation of chosen security countermeasures. This provides a means of highlighting any problems which may occur, which may then be overcome in advance of the actual implementation.

Finally, the Security Guidelines for Existing Healthcare Systems should not be considered in isolation and a number of the other SEISMED guideline deliverables are also relevant in the context of existing systems. These include specific guidelines relating to high-level security policy, system development and implementation, network security and data encryption.

7 Potential Problems

Whilst the new recommendations are intended to provide a simple and straightforward means of addressing healthcare security issues, it is recognised that problems may exist.

Firstly, many establishments may currently be operating with security significantly below the recommended level and progression to the required level may be a non-trivial task. As mentioned in the discussion of implementation, HCEs may face a number of constraints that affect their ability to address security requirements. For example, cost (in terms of finance, performance and practicality) will be a significant factor in determining acceptability. Financial cost will be particularly relevant, given that expenditure for direct care activities is likely to receive higher priority than security. In addition, organisational constraints will play a role insofar as recommendations will need to integrate with existing practice (or at least not conflict too greatly) in order to gain acceptance. If such constraints are present, establishments should bear in mind that every guideline implemented will improve the security of their systems.

Conversely, some environments and / or applications may demand a level of security significantly higher than the proposed baseline. In these cases a risk analysis review is recommended in order to determine the level of additional protection that is necessary. A specifically designed healthcare protection methodology, that has also been developed by this group, could be utilised for this purpose [8].

8 Conclusions
In conclusion, it is believed that the guidelines have fulfilled the objectives of this phase of the SEISMED project and will provide a solid foundation for the improvement of security within existing HCE systems.

Whilst the principles will remain relatively static, it is expected that the underlying guidelines will require periodic updates to account for changes within the healthcare field or in the types of information system technology available (e.g. the increasing use of multimedia systems may introduce new considerations). Changes within the local HCE (e.g. organisational structure, medical applications and practices) may also necessitate re-evaluation of some recommendations.

The guidelines will now form the basis of a further SEISMED workpackage dedicated to the validation of the projects recommendations. This will include full trials of the guidelines at the Reference Centres and will provide an extensive test of their applicability in practice. It is anticipated that the Reference Centres themselves will then be able to document their findings in due course.

Acknowledgements

We would like to acknowledge the various partners and collaborators within the SEISMED project for their valuable contributions during the development of these guidelines. In particular, we would like to thank the following individuals for their help and assistance throughout the project:

Dr Barry Barber (NHS Information Management Centre, United Kingdom);
Mr John Fowler (Royal London Hospital, United Kingdom);
Dr Nick Gaunt (Plymouth Health Authority, United Kingdom);
Dr Kees Louwerse (Leiden University Hospital, The Netherlands);
Prof. George Pangalos (University of Thessaloniki, Greece).

References


Abstract

The increasing use of and reliance upon information technology within modern healthcare establishments underlines a need for adequate security controls to protect the confidentiality, integrity and availability of systems and data. Whilst the consideration of security is now generally accepted as part of the design and implementation of new systems, many systems are already in operation in which these needs have not been adequately addressed. This paper presents a summary of the recommendations arising from the AIM SEISMED (Secure Environment for Information Systems in MEDicine) project relating to the addition and enhancement of security in existing healthcare systems.

The paper is based upon material originally presented at the SEISMED Workshop ‘Security and Legal Aspects of Advanced Health Telematics’, Brussels, 11 July 1994. The content has been revised in light of the workshop discussion and the further development of the guidelines since that time.

1 Introduction

The adoption of information technology has had a significant effect upon modern healthcare establishments (HCEs). Information systems are now utilised in most aspects of HCE operation, affecting areas from administration through to direct clinical care. It is, therefore, likely that healthcare professionals will become increasingly more reliant upon such systems to support routine working practices. As such, there is an increasing need for security controls to preserve the confidentiality, integrity and availability of systems and data.

However, a significant proportion of operational healthcare information systems were originally implemented without security needs having been properly addressed. This point was underlined by the results of the Survey and Risk Analysis investigations conducted within the SEISMED project, both of which revealed significant variety in the types of information systems in use and the levels of security currently provided.

As a result, methods are required by which security may be added or enhanced in these scenarios without rendering the systems unusable or uneconomic.

* Presented at SEISMED Workshop by Dr Barry Barber, NHS Information Management Centre, Birmingham, United Kingdom.
2 Security in Existing Healthcare Systems

The security of existing systems was addressed within SEISMED by workpackage SP07, "Security in Existing Operational Systems". The stated objectives of the workpackage from the outset of the project were as follows:

- to provide guidelines on the level of protection that should be attached to existing operational health care systems;
- to provide guidelines as to how the appropriate level of security in existing systems may be achieved;
- to revise the approach based on Reference Centre feedback.

The final deliverable of the workpackage was a comprehensive set of guidelines for adding security into existing systems.

2.1 Approaches for securing existing systems

As a result of discussions within the project two approaches were determined for addressing the security of existing systems, as shown in figure 1.

It is proposed that baseline guidelines should be considered in all cases, with protection profiles being utilised in especially sensitive scenarios. Both approaches are outlined in the sections that follow.

3 Baseline Security Guidelines

An early realisation was that the nature of the healthcare environment demands a standard (or baseline) level of security that is considerably higher than in many other fields.

Whilst various guidelines already exist for IT security in general, none have specifically targeted the medical community at a European level. A healthcare-specific baseline was, therefore, developed in close consultation with the SEISMED Reference Centres (and other independent healthcare professionals) to satisfy the following aims:

- to represent the minimal acceptable standard for security in healthcare establishments;
• to be usable by all HCEs and staff;
• to allow straightforward validation of existing systems against the baseline to ensure compliance.

Several interim generations of guidelines were produced during the course of the project for consideration by the Reference Centres. These were progressively refined and enhanced until a suitable set of final recommendations were produced [1].

3.1 Guideline Content

It was established from the outset of the work that, in order to provide comprehensive protection, any security guidelines for host systems would need to address more than just recommendations relating to a system in isolation. As such, the inclusion of guidelines for physical protection, personnel security and policy measures were all considered equally important.

On this basis a total of 138 guidelines for host systems security were developed. These are logically grouped according to 10 key principles of security as listed in table 1 below. The principles were chosen to represent the main elements governing the security of existing healthcare information systems.

<table>
<thead>
<tr>
<th>Principle Code</th>
<th>Title</th>
<th>Number of Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESP0100</td>
<td>Security Policy and Administration</td>
<td>5</td>
</tr>
<tr>
<td>ESP0200</td>
<td>Physical &amp; Environmental Security</td>
<td>22</td>
</tr>
<tr>
<td>ESP0300</td>
<td>Disaster Planning &amp; Recovery</td>
<td>7</td>
</tr>
<tr>
<td>ESP0400</td>
<td>Personnel Security</td>
<td>8</td>
</tr>
<tr>
<td>ESP0500</td>
<td>Training and Awareness</td>
<td>6</td>
</tr>
<tr>
<td>ESP0600</td>
<td>Information Technology Facilities Management</td>
<td>31</td>
</tr>
<tr>
<td>ESP0700</td>
<td>Authentication &amp; Access Control</td>
<td>28</td>
</tr>
<tr>
<td>ESP0800</td>
<td>Database Security</td>
<td>21</td>
</tr>
<tr>
<td>ESP0900</td>
<td>System Maintenance</td>
<td>5</td>
</tr>
<tr>
<td>ESP1000</td>
<td>Legislation Compliance</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 1: Principles of Existing Systems Security

It should be noted that the guidelines for database security present a sub-set of the information contained in the independent Database Security deliverable [2].

The underlying guidelines from each principle describe the concepts and countermeasures that should be adopted in order to achieve the recommended baseline security level. Coverage ranges from general concepts to specific technical issues.

3.2 Target Audiences

With such a broad coverage of areas it is to be expected that much of the material will
be inappropriate to certain categories of staff. As such, three distinct sets of guidelines have been developed as shown in figure 2, each targeting a different audience within the HCE.

Security Guidelines

![Diagram of Security Guidelines]

**Fig. 2**: HCE Target Audiences

However, the guideline sets are not totally independent and it is expected that the Management and IT & Security Personnel audiences will also read the General guideline set. Therefore, any General guidelines that are also applicable to the more specialised audiences are not duplicated in the other documents (unless new details have been added that are specific to the audience in question).

The focus of each guideline set is as follows:

- The General guidelines are aimed at the majority of HCE staff, including clinicians, administrators and general information system users. It is envisaged that a summary of the main points would be provided to staff as a basis for general day to day reference (50 guidelines).

- The Management guidelines primarily target the senior decision makers within the HCE, who will be responsible for defining security policy (61 guidelines).

- The IT & Security Personnel set is aimed at personnel such as IT staff, system administrators and security officers who will be responsible for implementing security (122 guidelines).

As an illustration of how the emphasis is altered for each target audience, the following example guidelines (all relating to the need to formulate and observe a Security Policy Document and taken from the Security Policy & Administration principle) may be considered:

"All users should acquaint themselves with the HCE security policy and observe any general regulations as well as any that may specifically apply to their role or department."
"Written documentation detailing HCE security policy (or a synopsis of the main points) must be available to all personnel. It should contain a clear definition of information security, as well as a clear and unambiguous explanation of the objectives and scope in relation to the HCE. The specific principles and guidelines implemented by the HCE should also be detailed."

"Technical staff should provide relevant expertise to assist management in the formulation of the HCE security policy. They should subsequently acquaint themselves with the policy in full and observe any general regulations as well as any that may specifically apply to their role or department."

4 System Classification and Profiling

Whilst the recommendations presented by the guidelines are considered to be comprehensive, it is envisaged that the baseline level of protection may not be sufficient for some sensitive healthcare application areas. As a result, it will be necessary for the further security requirements of these HCEs / systems to be established on an individual basis.

However, some establishments may consider a full risk analysis study for each individual system to be prohibitive in terms of both financial cost and time. This highlights the requirement for a simplified means by which additional security needs can be identified in such cases.

A potential approach identified within SEISMED is for existing systems to be classified according to predetermined "protection profiles", selected via an accompanying methodology [3]. The basis for such an approach is the classification of existing healthcare information systems using an appropriate combination of the key elements listed in figure 3.

Fig. 3 : Elements of Existing System Profiles

It is possible to divide each of these elements into a number of further sub-categories. For example, the computer configuration is assessed on the basis of whether the machines involved are desktop PCs, portables or mini / mainframe systems and whether any networking aspects are involved. Protection related factors of the operational environment include the physical location, the nature of the buildings in which the system is housed and, finally, the number and mixture of people involved. Data sensitivity is assessed on the basis of the types and uses of data in the host system.
A series of generic categories for both factors are included in the methodology (in the form of a healthcare generic data model), with associated impact ratings for information disclosure, denial, modification and destruction.

The configuration and environment elements are considered to determine the basic protection requirements of a system (i.e. regardless of how it is actually used), with the assessment of data sensitivity building upon this to complete the profile. A series of predetermined profiles would exist for each element type (e.g. computer configuration profiles for personal, portable and mainframe systems), detailing the countermeasures required to deliver protection at different levels of sensitivity. These can be combined to represent many typical HCE information system scenarios and thereby describe an appropriate set of overall security countermeasures.

In practice, the main stages involved in applying the methodology would be as follows:

- determine basic system profile by identifying configuration and environment;
- assess data sensitivity from the types and uses of data in the existing system;
- determine overall protection profile and associated countermeasures;
- select and implement final countermeasures.

This last stage would be tempered by any HCE-specific factors which might limit the suitability or acceptability of the recommendations. For example, cost constraints, operational overheads and staff culture would all need to be considered before final countermeasure selection.

A more detailed description of the classification methodology is presented in the referenced article and in the context of a supplementary deliverable from the workpackage.

5 Conclusions

It is believed that the final guidelines will serve to provide a comprehensive source of reference for European HCEs in relation to the protection of existing medical systems. The protection principles that have been established should remain relatively static and will be applicable, to some degree, in virtually all scenarios. It is, however, anticipated that the underlying guidelines may require future revisions to take account of changes in healthcare IT practice.

It is acknowledged that many HCEs / systems may currently be operating with a level of protection significantly below the recommended baseline and may be faced with financial and / or organisational constraints that will complicate the process of change. As such the adoption and enforcement of the guidelines may represent a non-trivial task. However, it should recognised that each measure implemented will improve the overall security status of the HCE.

In terms of further development, the guidelines have already formed the basis of a Validation workpackage, to ensure their applicability to the full European healthcare
community. They have also been utilised in a SEISMED training programme to increase security awareness within European establishments.

It is hoped that, in combination with the other guideline deliverables from the project, the guidelines for existing systems security will help to achieve a harmonised framework for the protection of healthcare establishments and systems throughout Europe.

References

CONTAINS DISKETTE
UNABLE TO COPY
CONTACT UNIVERSITY
IF YOU WISH TO SEE
THIS MATERIAL