1 Introduction

🗀 http://www.asecuritysite.com/security/information/chapter01

1.1 Objectives

The key objectives of this unit are to:

- Provide an overview of some key terms, such as CIA and AAA.
- Provide an overview of current systems and infrastructures, such as with Cloud Computing.
- Define some key principles, such as defence-in-depth and de-militarization zones (DMZ).

1.2 The Industrial and the Information Age

In a matter of a few decades the World has changed from an industrial age into an information age. It is one which, unlike earlier ages, encapsulates virtually the whole World. It is also one which allows the new industries to be based in any location without requiring any natural resources, or to be in any actual physical locations. Typically all that is required is a reliable network connection (Figure 1.1).

Our world is changing by the day, as traditional forms of business are being replaced, in many cases, by more reliable and faster ways of operating. Our postal system, while still used for many useful applications, has been largely replaced by electronic mail. With voting, the slow and cumbersome task of marking voting papers with the preferred candidate, is now being replaced by electronic voting. The traditional systems, though, have been around for hundreds if not thousands of years, and typically use well tried-and-tested mechanisms. For the most part, for example, we trust a paper-based voting system, even though it is well known that a count of the votes within an election will often produce different results each time that the vote is counted, and then recounted. An electronic method will, on the other hand, most likely have a success rate of 100%.

As for paper-based voting, most countries just require a simple form of authentication, such as a printed piece of paper which contains the name and address of the person, which could, of course, be simply printed by an ink jet printer. The electronic form, with, at a minimum, a unique user ID and personal password is more verifiable than this, but still many people think that the paper-based method is more secure. At the core of this misunderstanding is that the existing system of using the Internet is flawed, in that its applications, protocols and communications are open to abuse, thus the future of the Internet, and the applications that it supports, require much more assurance in every part of the communications and also some measure that the data is secure in as many ways as possible. One flaw in this process can often bring the whole system into question. This book aims to present an outline of the protocols and systems used, and will hopefully allow the reader to assess these and to evaluate their operation and, above all, their validity. The book is divided into two parts and

focuses on the different aspects of security, from its implementation in host and network device to software integration, and on the way that the data can then be used to investigate incidents using forensic computing methods.

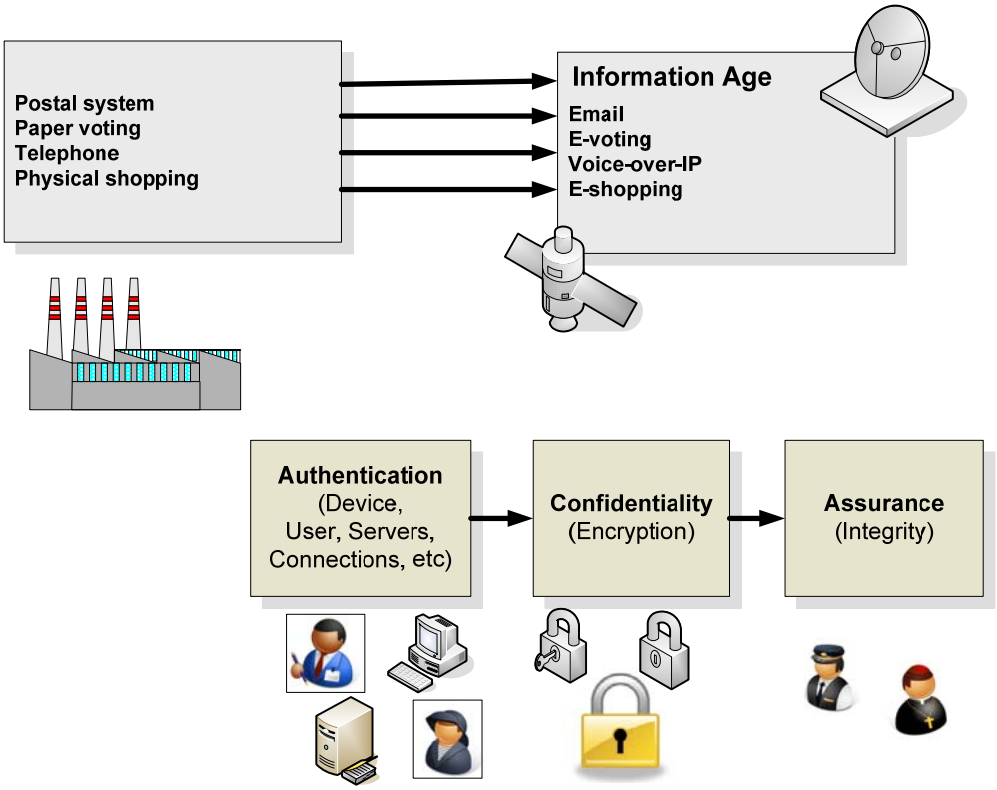

Figure 1.1 From an industrial age to an information age

1.3 CIA and AAA

A major problem in security is that systems tend to be created in a layered approach, such as: the applications and services; the application/network protocols; and the network infrastructure. Unfortunately, the interconnect of these tends to be one of the weakest points in the systems, in terms of security. Figure 1.2 outlines this, along with two important concepts in security which are: CIA (Confidentiality, Integrity and Assurance); and AAA (Authentication, Authorization and Accounting). With **confidentially** the entity, whether it be for information or a device, should only be viewed by an authorized entities (such as a priviledged user). For **integrity** it is important that systems do not corrupt any of the data, and must guard against unauthorised, malicious or accidental data changes. Thus integrity requires that all actions must be authorized, and that each entity has a unique security policy, or one which is inherited from other entities. It may also have some form of error detection/correction to prove the integrity of the data – and thus provide **assurance**.

As the requirement for highly available systems increase, there is a need for **availability** where the entities within the system must be available in a usable form, and must give the required quality of service, within limits such as responsiveness, usability, and so on.

A worrying trend in network systems is for devices or users to be *spoofed*, such as where an intruder *steals* a valid user ID and password, and uses it as a foothold into a system. Another problem can occur where access to a network is based on the physical network address of a host. This, unfortunately, can also be easily spoofed. Thus a key element of enhanced security is **authentication**, which can be based on many things, such as: a user ID and password; a digital certificate; a process ID; or the physical address of the network adapter. Once the identity has been verified, the access to networked services must be **authorized**, of which there can be also be some form of **accounting**. AAA is a model used for network security, where many network access devices, such as wireless access points, can implement some form of AAA.

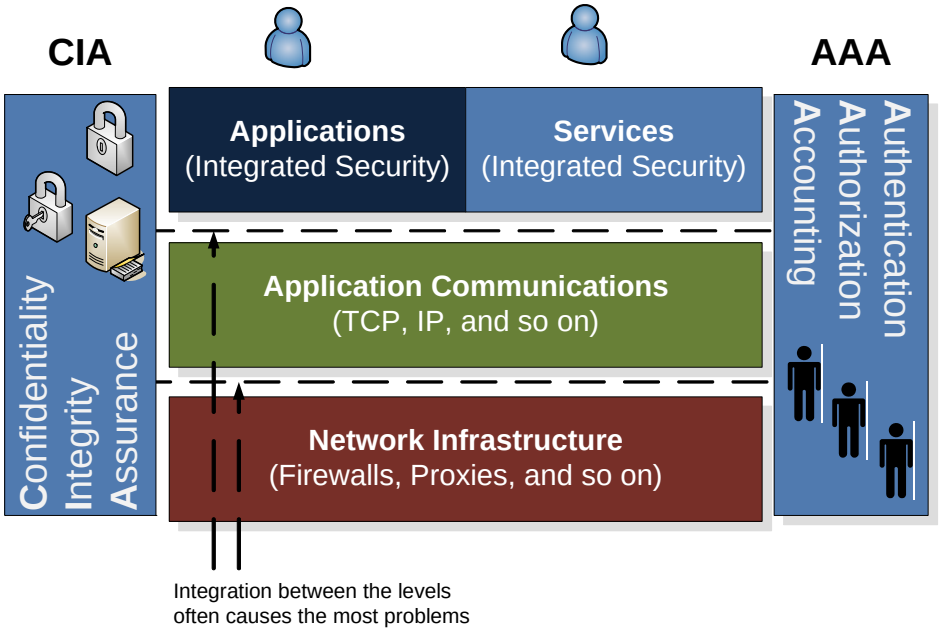

Figure 1.2 CIA and AAA

1.4 Protecting against intruders

There is generally a perception that systems must be protected against individuals who aim to do damage to them, but this is not often the case, as this type of intrusion can be easily overcome. The major problem often comes from concerted attacks from teams of individuals, often with large budgets. As Figure 1.3 illustrates, it becomes more difficult to cope with an intrusion as the budget increases. The various levels are:

- **Home user**. At the lowest level, the user at home will typically have a limited budget for their activities for both hardware and software, and will often have very little opportunity to actually physically intrude into the network.
- **Data miner/hijacker**. Data mining is becoming an important source of information for a wide range of interested parties, as they may wish to gain information on users and groups of individuals. For this a professional data miner may have a reasonably large budget in which to steal data, and spy on individuals. The increasing use of malware and spyware shows how dangerous this type of activity is becoming, where small programs are installed on host ma-

chines, which can then steal data, or determine the Web pages that a user is most likely is to go to, and can actually even change the results given from a Web search. Organisations are also under attack from this type of activity where external intruders can gain information from inside a network.

- **Industrial espionage**. This type of intrusion can obviously have a relatively large budget, as data gained from other organisations can save a great deal of time in data gathering, and also in research and development. For example a commercial company may be able to gain a competitive advantage, if it steals the designs of a new product from another company, or, in some way, manages to corrupt the design files of a rival company.
- **Government activities**. Governments have key security issues, such as protecting themselves against attacks from both internal and external interests. They must, thus, have a surveillance plan which gleams data from various places in order to keep up their defence activities. Included in this would be covert investigations by the police.
- **Large-scale military**. The largest budget is typically achieved with military operations, which are likely to have vast budgets to gain data from users and organisations. Unfortunately, from an organisational point-of-view, it is extremely difficult to guard against this type of intrusion, but organisations must be aware of the possibilities.

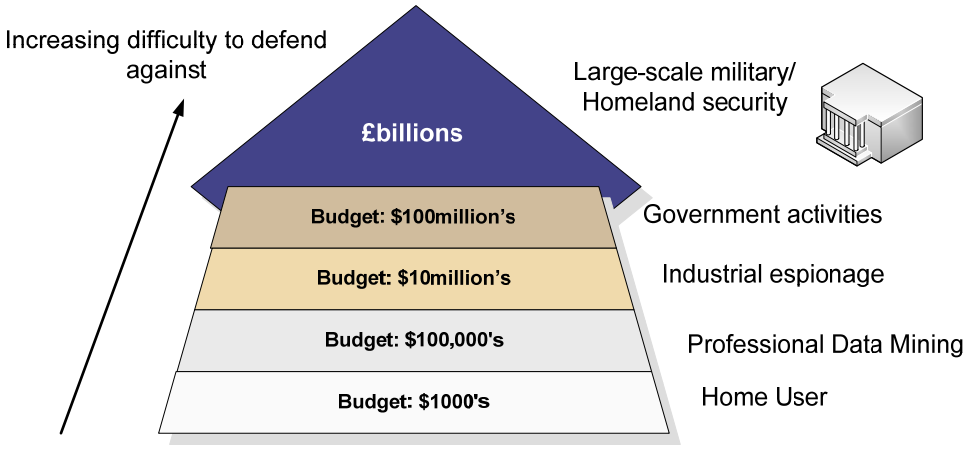

Figure 1.3 Possible budgets for intrusion

1.5 Users, systems and data

Information security is becoming one of the most important aspects for many organisations. This is due to many reasons, from fulfilling their social and moral duties, to preserving their ideas and intellectual property. In general, organisations must protect against from external intruders, and also from internal ones. A major problem is that many of the defence mechanisms are setup to guard against external attacks, but it has been shown that many attacks originate from inside a network.

A defence system must protect (Figure 1.4):

- **Users**. Organisations have a legal and moral responsibility to protect their employees from abuse, either from inside the organisation, or from external sources. Examples of this might include protecting users from objectionable content, and from spam emails.
- **Systems**. A key element of any defence strategy is to protect systems from both internal and external attacks. This includes the actual physical protection of the devices and also their configuration, along with their software and hardware components. Any weaknesses can be costly in terms of time and money to fix a system which has been breached.
- **Data**. The protection of data is important for many reasons, as a loss or change of data can be expensive to any organisation. A key element of any data is to safeguard its confidentially, its integrity and to have some assurance that it has not been tampered with. This leads to the **CIA** (Confidentially, Integrity and Assurance) principle which are the guiding principles for data security.

There are a whole host of threats that organisations face, including its users, its systems and its data. In order to simplify the protection of the overall system, it is typical that an organizational network has a single gateway, which provides the main flow of traffic into and out of the network. This is, thus, the main bastion and allows for traffic flow to be monitored and for a quick reconfiguration in the face of an external attack. Figure 1.4 shows an illustration of the gateway which runs a firewall to block unwanted traffic from outside the trusted system. Examples of the threats include data stealing, worms and viruses, denial-of-service, and so on. On the trusted side, the organization wants to provide corporate access to external systems, as well as providing the required services to its users, such as email and Web access. Threats from outside are obviously a major problem, and firewalls close to gateways are used to reduce the threat. These firewalls filter the network packets by examining their contents, and deciding whether to drop them, or not. Unfortunately firewalls can often be easily breach, such by an external intruder, or by a virus, which arrives in another application program, such as in an email attachment. The major problem is thus to protect the trusted system from the inside, as illustrated in Figure 1.5.

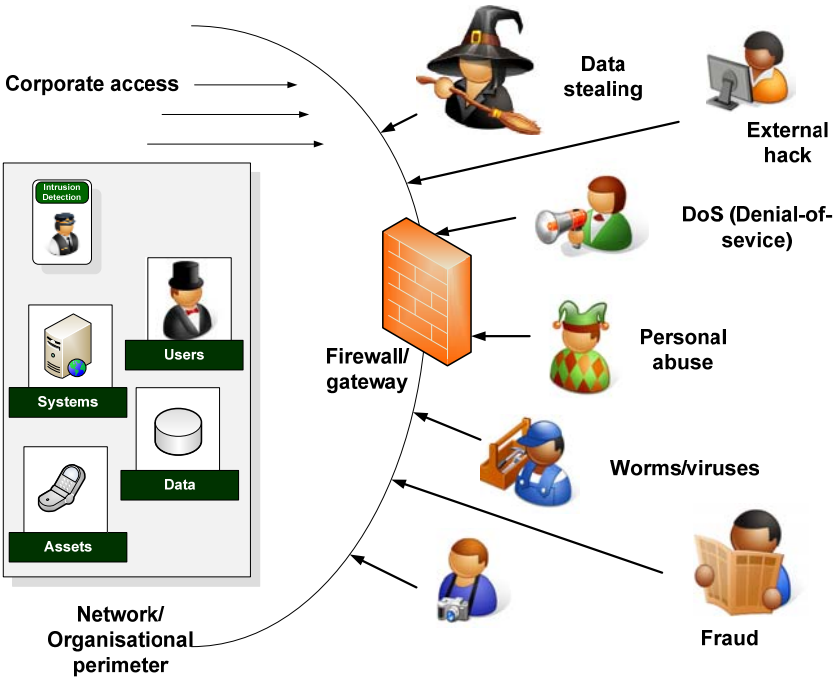

Figure 1.4 Internal and external threats

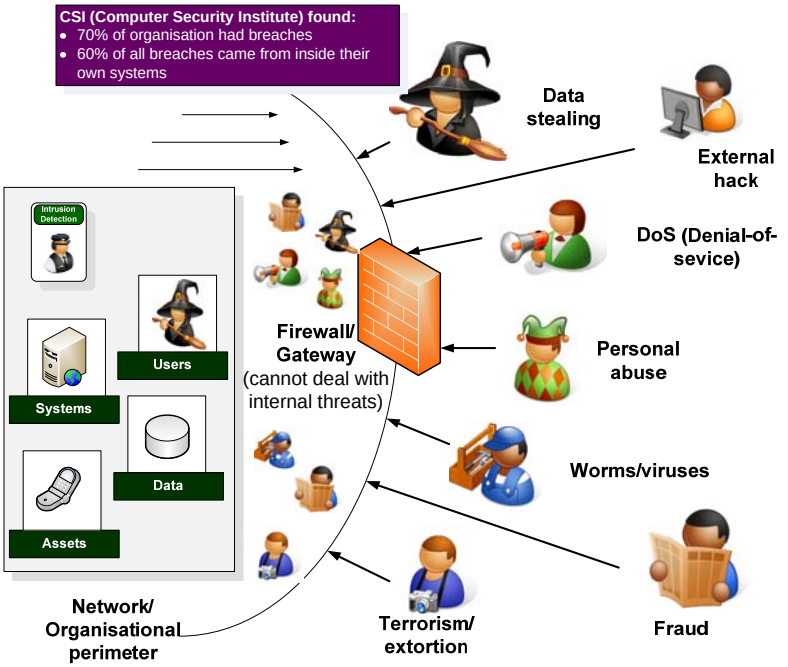

Figure 1.5 Internal and external threats

1.6 Services, role-based security and cloud computing

The Internet and networks have evolved where organisations typically designed, created and maintained their own customized networks which provided a number of services to their users. These services include electronic mail, Web servers, remote access, and so on. The cost of this for many organisations is becoming difficult, and there is generally a growth in external organisations providing these services, for

which the organisation can subscribe to. This often reduces the exposure of the organisation to certain risks, and thus they do not require localized expertise to gain access to certain services. This concept of subscribing to external services is known as **cloud computing** (Figure 1.6), where services are provided from an external provider (*within the cloud*).

The need for robust authentication will thus become more important, especially as it is linked to roles within an organisation. For example, an organisation may have an external authentication provider to authenticate a user, and this authentication will then be linked to the rights they have to run certain applications on the network. For example, a medical doctor might authenticate themselves against a national database of medical practicioners, and then receive the rights based on their role and identity to access a medical database, and also access to the patients for which they are currently associated with. This provides us with the concept of **role-based security**, where the rights that users have is based on their role within an organisation, or on external roles. These internal roles will be well-known within the organisation, such as surgeons, nurses, GPs, patients, and so on, within a health care environment, and external roles would be defined between the interfaces between different domains. Figure 1.7 outlines two domains (health care and education), and shows that roles can be identified within each of the domains, and then the rights assigned based on this. A role-based approach is often well understood within organisations. It also makes it easy to add users to and delete from certain roles. One approach that organisations can take is to use a least-privilidge concept for their roles, where the role get a minimum set of priviligees for their role. This is often highly secure, but users often feel that they are too constrained in what they can do on the system. For example in a health care environment, a GP might need emergency access to a patient's records, of which they are not assigned too. These can often, though, be dealt with on an exception basis, where in certain conditions it becomes allowable for users to override their privileges, but that these expections are logged, along with the reasons for them to be overruled.

Another balance is between complex security solutions against simple ones, where complex ones are often difficult to manage, especially in the face of a threat, whereas simple ones are often easier to modify, and to understand.

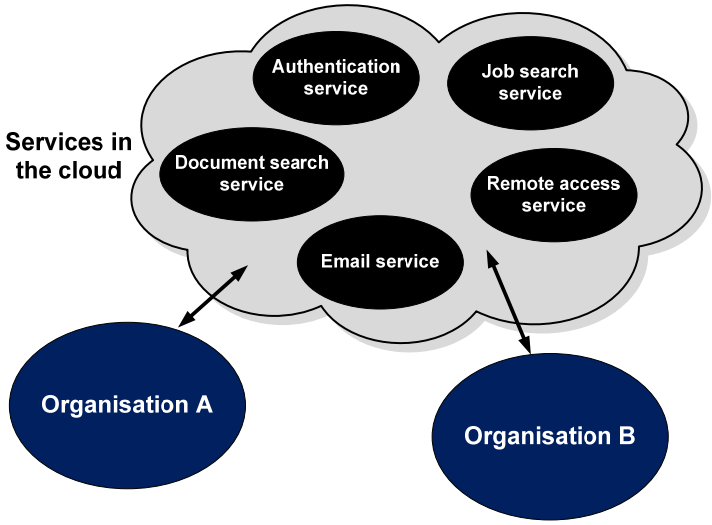

Figure 1.6 Services in a cloud

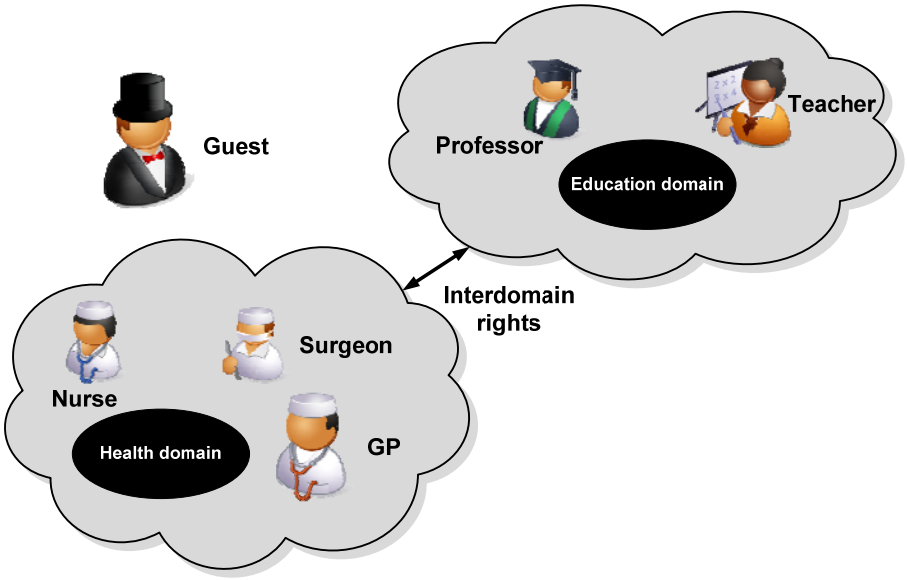

Figure 1.7 Interdomain rights

1.7 Security and Forensic Computing

Information security is typically viewed as the protection of information from being viewed or changed by those who do not have the rights to it. This can include the transmission, storage and/or the processing of the information. As these systems often involve data storage, processors, processes, memory, users, network connections, network protocols, and so on, it is thus a difficult task to make sure that the complete system is operating securely. Forensic computing, on the other hand, is the collection, preservation, analysis and reporting involved in an investigation which involves some form of electronic/computer communication and/or computer related activity. Figure 1.8 outlines the relationship between security and forensic computing, where a security policy is defined within the corporate infrastructure and also onto the interconnected hosts. This policy might relate to the network traffic that is allowed to flow into and out of the network, and the rights of users/groups.

Normally, to simplify the control of an IT system within an organisation, there is a single entry/exit point, known as a gateway or perimeter device. The security policy is then defined within this perimeter. It may also define the data that will be logged on networked devices, and, possibly on hosts. This might include recording all the accesses of hosts to remote Web sites, or the recording of the applications that were run on certain hosts on the network. The aim of these logs might be to provide some form of accounting of the services on the network, but could obviously be used for some future forensics purpose.

The detection of malicious activity, or some form of activity which breaches the operation of the system, can be achieved with event detectors, which are used to generate event messages (which are typically known as alerts). These events, though, will typically be false alerts, and will not be acted upon, but, at times the events will lead to some form of forensic computing investigation, which will then involve the collection of data for the investigation, and finally onto some form of report. It is thus key that that the forensics activities should be thought of at an early stage, so that the required data can be gathered. For example, a bank might define certain events which define corporate fraud, such as detecting when a user tries to login with more than ten times for a certain user ID, or if they pay for goods worth several thousands of dollars, where they have only ever paid a few dollars for goods in the past. Another application of event detection is in intrusion detection systems (IDSs) where malicious activity has been detected in the past, and a signature of this activity is defined.

The requirement of forensic computing is thus many fold and might include investigating:

- **An intrusion on a system**. This might lead to a criminal prosecution, but most of the time the intrusion is investigated in order to be able to detect it in the future, and to overcome it an early stage.
- **A criminal activity.** This might lead to a criminal prosecution, or to thwart the activities in the future.
- **Breach of security policy.** This might lead to a disciplinary procedure within an organisation.

There are three levels of laws in most countries: criminal; civil and administrative. **Criminal Law** normally involves some form of criminal investigation which involves a criminal act, and which might result in fines or imprisonment, whereas **Civil Law** (or Tort Law) relates addressing wrongs that have been done (and are outwith contractual arrangements). This might relate to someone distributing copyrighted material without the creator's permissions. Whilst administrative law, relates to the enforcement of goverment regulations. An example of this is where a company has to make sure that it switches-off all its computers which are unused between 8pm and 5am, and a failure to do so may result in a fine.

Another important concept in the legal aspect of security and forensic computing are the concepts of **Due care** and **Due diligence**. With due care, the organisation must

make sure that it has taken the correct steps in the creation and implementation of its security policy and in its risk analysis. Then due diligence relates to the actual operation and maintance of its security system, especially for vulnerability testing. Thus a company should take due care in analysing and designing their security policy, but not take due diligence in actually proving that it works. It can work the other way, in that a policy might be implemented with due diligence, but the orginal creation of the policy has not been properly analysed/designed. It is thus important, in terms of any future liability, that security systems are designed, analysed, implemented and maintained with both due care and due diligance.

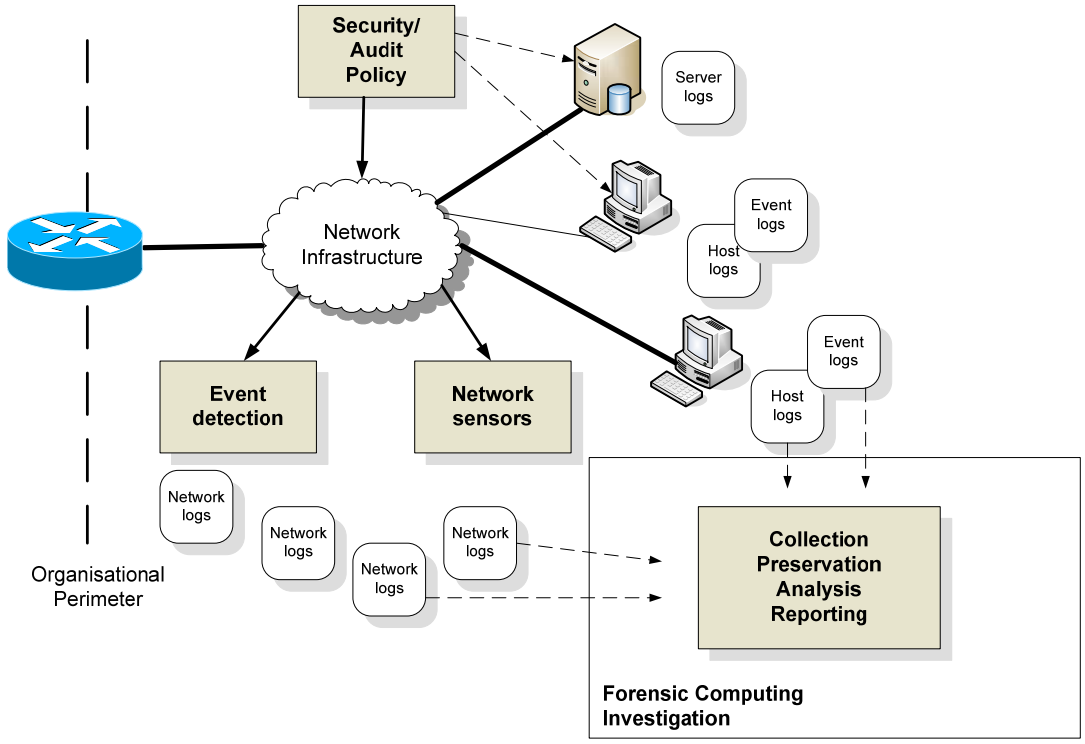

Figure 1.8 Information security and forensic computing

1.8 ISO 27002

The ISO 27002 standard started life as "Information Security Code of Practice" from the UK (DTI), and was published in the 1990. It recently changed from ISO/IEC 17799 to ISO/IEC 27002 and provides a benchmark for most areas of security. Overall it defines 11 main areas:

1. Business Continuity Planning.

To counteract interruptions to business activities and to critical business processes.

2. Access Control

- Control access to information.
- Prevent unauthorised access to information systems.
- Ensure the protection of networked services.

- Prevent unauthorized computer access.
- Detect unauthorised activities.
- Ensure information security when using mobile computing and tele-networking facilities.

3. System Acquisition, Development and Maintenance

- Ensure security is built into operational systems.
- Prevent loss, modification or misuse of user data in application systems.
- Protect the confidentiality, authenticity and integrity of information.
- Ensure IT projects and support activities are conducted in a secure manner.
- Maintain the security of application system software and data.

4. Physical and Environmental Security

- Prevent unauthorised access, damage and interference to business premises and information.
- Prevent loss, damage or compromise of assets and interruption to business activities.
- Prevent compromise or theft of information and information processing facilities.

5. Compliance

- Avoid breaches of any criminal or civil law, statutory, regulatory or contractual obligations and of any security requirements.
- Ensure compliance of systems with organizational security policies and standards.
- Maximize the effectiveness of and to minimize interference to/from the system audit process.

6. Human Resource Security

- Reduce risks of human error, theft, fraud or misuse of facilities.
- Ensure that users are aware of information security threats and concerns, and are equipped to support the corporate security policy in the course of their normal work.
- Minimise the damage from security incidents and malfunctions and learn from such incidents.

7. Security Organisation

- Manage information security within the Company.
- Maintain the security of organizational information processing facilities and information assets accessed by third parties.
- Maintain the security of information when the responsibility for information processing has been outsourced to another organization.

8. Computer and Network Management

- Ensure the correct and secure operation of information processing facilities.
- Minimise the risk of systems failures.
- Protect the integrity of software and information.
- Maintain the integrity and availability of information processing and communication.
- Ensure the safeguarding of information in networks and the protection of the supporting infrastructure.
- Prevent damage to assets and interruptions to business activities.
- Prevent loss, modification or misuse of information exchanged between organizations.

9. Asset Classification and Control

- Maintain appropriate protection of corporate assets and to ensure that information assets receive an appropriate level of protection.

10. Security Policy

- Provide management direction and support for information security.

11. Security Incident Management

- Anticipating and responding appropriately to information security breaches.

12. Risk Analysis

- Understand risks involved.

1.9 Risks

Threat analysis is a growing field, and involves understanding the risks to the business, how likely they are to happen, and their likely cost to the business. Figure 1.9 shows a plot of cost against the likelihood, where a risk with a likely likelihood, and low costs, is likely to be worth defending against. Risks which are not very likely, and which have a low cost, and also a risk which has a high cost, but is highly likely, are less likely to be defended against. At the extreme, a high risk which has a low likelihood and which has high costs to mitigate against is probably not worth defending against. The probabilities of the risks can be analysed using previous experience or from standard insurance risk tables. Figure 1.10 outlines an example of this.

1.9.1 Single Loss Expectancy/Annual Loss Expectancy

One method of understanding the cost of risk, it to determine the single loss expectancy, which is calculated from:

ALE = AV x ARO

Where:

ALE is the Annual Loss Expectancy

ARO is the Annualized Rate of Occurance.

AV is the value of the particular asset.

For example if the likelihood of a denial-of-service on a WWW-based database is once every three years, and the loss to sales is £100K, the ALE will be:

ALE = £100K x 1/3 = £33K per annum

This formula assumes that there is a **total loss** for the asset, and for differing levels of risk an EF (Exposure Factor) can be defined as the percentage of the asset damage. The formula can then be modified to:

ALE = AV x ARO x EF

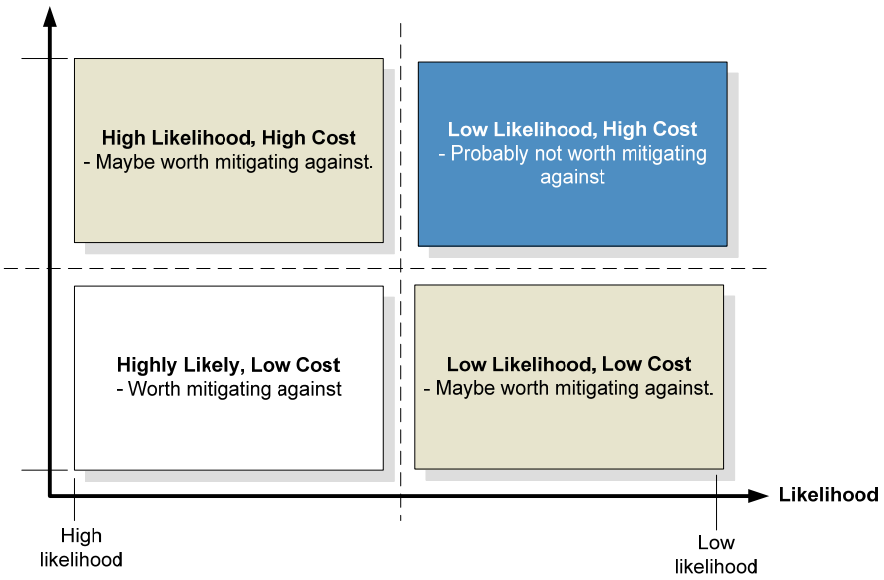

Figure 1.9 Risk analysis

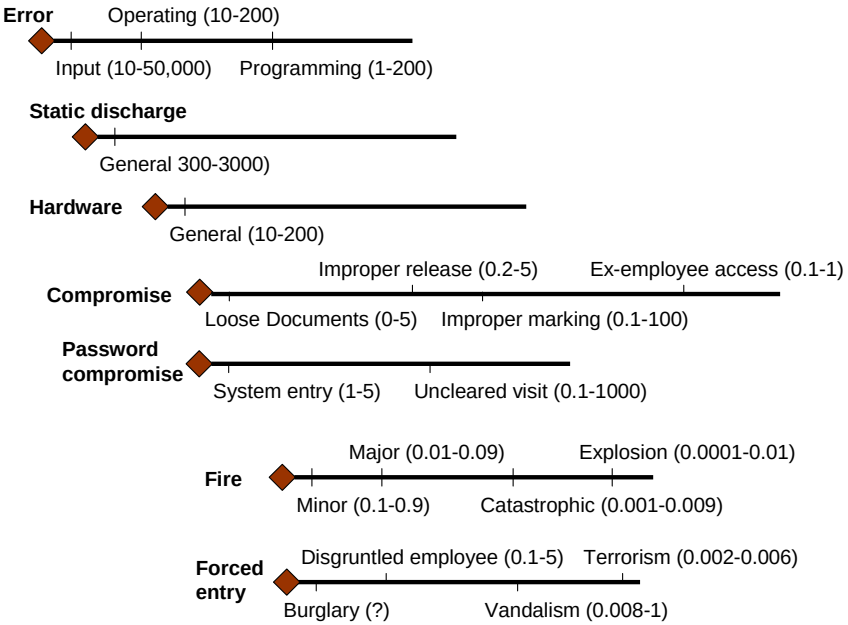

Figure 1.10 Risk analysis for various activities

1.10 Risk management/avoidance

The major problem in risk and in defining security policies is that there is often a lack of communication on security between business analysts and IT professionals, as they both tend to look at risk in different ways. Woloch (2006) highlights this with:

> "Get two risk management experts in a room, one financial and the other IT, and they will NOT be able to discuss risk. Each puts risk into a different context … different vocabularies, definitions, metrics, processes and standards …"

CORAS (A Framework for Risk Analysis of Security Critical Systems) is one system which has been developed to try and understand the risks involved, and to develop an ontology. This ontology (as illustrated in Figure 1.11) allows everyone to speak in the same terms. For example: A **THREAT** may exploit a **VULNERABILITY** of an **ASSET** in the **TARGET OF INTEREST** in a certain **CONTEXT**, or a **THREAT** may exploit a **VULNERABILITY** opens for a **RISK** which contains a **LIKELIHOOD** of an **UNWANTED INCIDENT**. In this way, all of those in an organisation, no matter their role, will use the same terminology in describing threats, risks and vulnerabilities.

For **risk management**, it is understood that not all threats can be mitigated against, and they must thus be managed. Figure 1.12 shows the methodology used by CORAS in managing risks, where a risk might be accepted if the cost to mitigate against it is too expensive. Network sensors thus then be setup to try and detect potential threats, and deal with them as they occur. For **risk avoidance**, systems are setup so that a threat does not actually occur on the network. An example of risk management is where a company might not setup their firewalls to block a denial-of-service (DoS) attack, as it could be seen as this might block legitamate users/services, and could thus install network sensors (such as for Intrusion Detection Systems) to

detect when a DoS occurs. With risk avoidance, the company might install network devices which make it impossible for a DoS attack to occur.

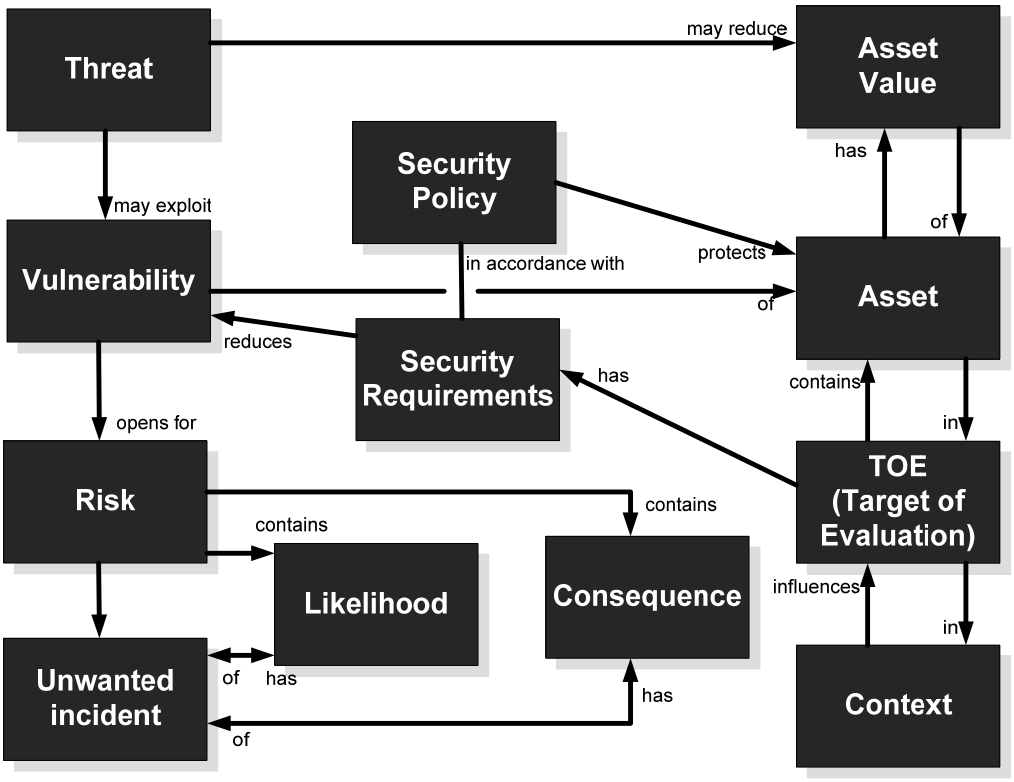

Figure 1.11 CORAS Ontology

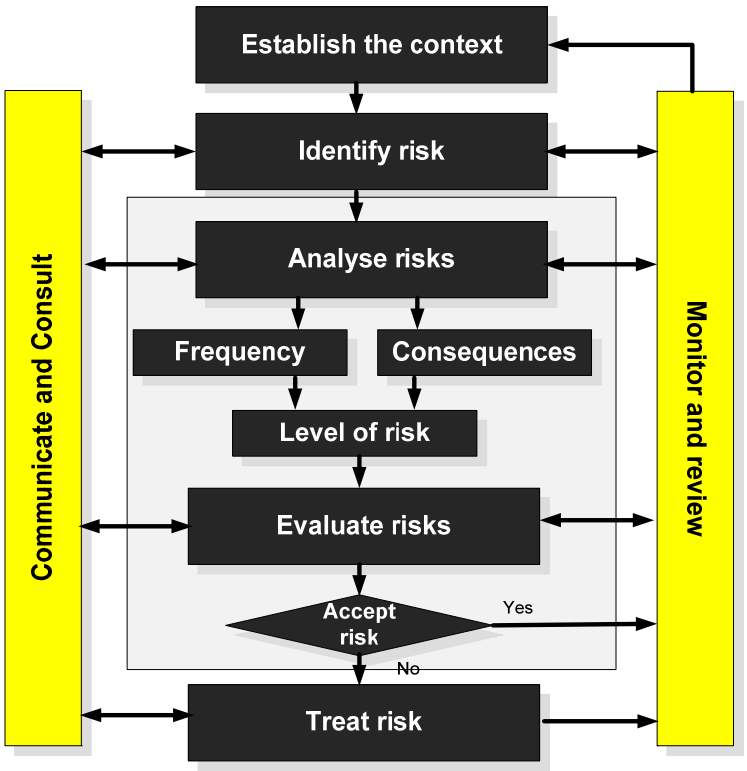

Figure 1.12 CORAS risk management

1.11 Security policies

Military analogies are often used in security, and the equivalent of having internal and external attacks is equivalent to fighting an external army at a defensive line, while also being attacked from behind a defensive line. In networked systems, with security, there is no attack, as the goal is purely to defend. Overall the key factor in any security system is that the aims and objectives of the organization must map directly onto the implementation of the security policy.

The security policy is often made up of multiple documents which focus on given audiences. As an overarching methodology the **Govering Policy** defines the highest level, and outlines the organisation's role with respect to security, and how users should interact with the policy. It should also highlight the relavence of the policy to their activities, especially on the needs to meet compliance obligations. This policy is mainly aimed at a managerial (as they must inform the staff under their control) and technical specialist (as they must actually implement the Govering Policy) level, and should be unambiguous in its definition of the overall strategy of the Govering Policy, such as:

> Employees will not save any music files on their file systems, including disk drives and memory sticks, such as MP3 and WAV, or within ZIP files, for which they do not have the correct rights to. This is required as the organisation must comply with existing copyright requirements (Ref: Copyright Law of the United States). All media of this type found will be logged, and delete. A report will be sent to the require Head of Unit for further action.

Below the Govering Policy are the **Technical Policies**, the **End-User Policies** and the more details policies, such as those related to Standard, Guidelines and Procedures. The Technical Policies typically relate to the technical implementation of security related to specific services, such as the technical policy relate to the usage of email. In this case it might define the operation of the spam filters, or the adding/deletion of spam email addresses. For the end-user policies, it is written in a way which the user understands their obligation on accessing network resources and services. It is normally written in a plain way, such as:

> Users who access the network must **not** leave their computers logged-in when they leave their desk. They should also not pass-on their username and passwords to any others, apart from when requested by a System Administrator. Failure to comply with this will result in a report being sent to the appropiate Departmental Manager for further investigation.
>
> Users should only use the Web for business activities, and they should understand that all accesses to external Web pages will be logged, and could be used for future purposes. Users who have been found spending more than 10% of their time on non-business activities will be reported to the appropriate Departmental Management for further investigation.

The user should then agree to this, or not. If they do agree they will be held to this. It is thus important that it is clear to all users of their obligations, and the results of any breaches.

There are also a whole host of documents which are more detailed and related to various standards, guidelines and procedures. The standards documents relate to the actual operation of systems and protocols, such as for ISO 27002 standard. With guidelines, as apposed to standards, there is no definitive practice, which an outline as to best practice. For example the NIST (National Institute of Standards and Technology) published many guidelines on security such as for SP 800-124 (Guidelines on Cell Phone and PDA Security) and SP 800-115 (Technical Guide to Information Security Testing and Assessment).

1.11.1 Security policy elements

A networked system is a complex entity, composed of many elements, such as hardware devices, operating systems, application programs, file systems, and users. In a highly secure system the overall system should be also be broken down into entities, each of which have security policies for individual users, and also for groups of users. For example, a networked printer should have a policy which restricts access to individual users, and also groups of users. Often in a hierarchal network the entities should inherit security policies from the hierarchy above them. For example with file directories, the subdirectories will often inherit their security policies from the level above, unless otherwise stated. This type of approach typically simplifies the security policy for the overall system.

Often the key elements of any security policy are to:

- **Deter**. This is where the system is designed and implemented in order to initially deter intruders from attacking the system in the first place.
- **Log**. This is a key element in modern systems which require some form of logging system. It is important that the data that is logged does not breach any civil liberties, and is in a form which can be used to enhance the future security of the system.
- **Detect**. This is where detection agents are placed within the network to detect intrusions, and has some method of tracing the events that occurred in an intrusion, so that it can be used either in a forensic computing investigation, and/or to overcome a future intrusion. Organisations often have many reasons for detecting network traffic, such as illustrated in Figure 1.5.
- **Protect**. This is where policies are created which protect systems, users and data against attack, and reducing this potential damage. A key element of this is to protect them against accidental damage, as accidental damage is often more prevalent than non-accidental damage.
- **React**. This is where a policy is defined which reacts to intrusions, and defines ways to overcome them in the future. Often organisations do not have formal policies for this type of activity, and often rely on *ad-hoc* arrangement, where the method of reacting to a security breach is created after the event.

- **Recover**. This is where policies are defined to overcome any system damage, whether it is actual physical damage, the abuse of users; or the damage to data.
- **Audit/verify**. It is important that the security policy allows for auditing and for the verification that it achieves its requirements.

Security, typically, focuses on the detection, protection and recovery from an attack, whereas forensic computing focuses on not just the malicious activity, but also in capturing the after-affects of an attack, as well as for non-malicious behaviour. A key component is that security tends to focus on the assumption of guilt within attacks, whereas forensic computing must focus on both malicious and non-malicious data so that a fair case can be presented for an investigation. Thus a forensics policy will typically focus on the detection of events, and the associated procedures. The key focus for the forensic computing parts of this book will be on:

- **Log**. This will define the data that is recorded, and, possibly, the rights of the data to be viewed by certain individuals within an organisation.
- **Detect**. This would be the activities which were to be detected for forensic investigations.
- **React**. This is where a policy is defined which reacts to malicious activities, and, especially in a forensic computing investigation, the procedures involved.
- **Audit/verify**. It is important that the forensics policy allows for auditing and verification that it achieves its requirements.

1.12 Defining the policy

A key element of security is to have a policy which is defined at the highest-level in the organisation, and which is well-known to the employees in the organisation. Also, if there is public access to the network, external users should also be informed as to the security restrictions placed on the network. Figure 1.13 shows a transparent and auditable system where the policy is defined at the highest level, and includes: the aims and objectives of the organisation; the legal moral and social responsibilities of the organisation; and the technical feasibility of the policy. These are then decided upon, and a policy is implemented by technical staff. A key feature is that this policy should be audited in some way, and also verified that it achieves the policy requirements.

There are many different types of network/user activity that should be detected and which could breach the aims and objectives of the organisation, or which breach the social, moral and legal responsibilities of the organisation. Examples of classifications for attacks might be:

- Attempted administrator privilege gain.
- Attempted user privilege gain.
- Denial-of-service.
- ICMP event.
- Information leak.

- Network scan.
- Non-standard protocol.
- Policy violation.
- Suspicious string detection.
- Suspicious login.
- Trojan activity.
- Unusual client-port connection.
- Web application attack.

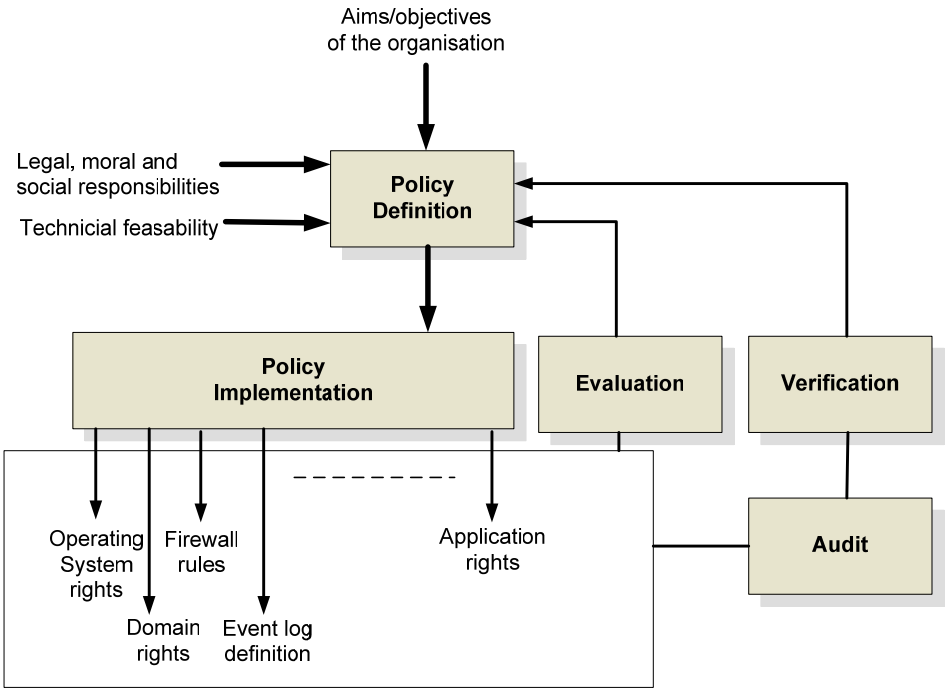

Figure 1.13 Security policy definition, implementation and verification

There are many examples of network traffic/user activity that might be monitored with an intrusion detection system (IDS). It can be seen that it is not just threats to the network, but also activities that might be wasteful in resources, or which breach social and moral rules. It can often be just as embarrassing for a user in an organisation to be involved in an immoral activity, than it is to have a network intrusion. Thus applications such as peer-to-peer file sharing, such as Kazaa, should be avoided in organisations, as they have many copyright issues. Along with this audio and video streaming, such as from news sites, may be wasteful on bandwidth, and, if this type of traffic was great enough, it might swamp traffic which is important for the organisation.

1.13 Example risks

There are many different types of attacks, including:

External misuse:

- **Visual spying**. This is the actual physical viewing a user's activities, such as their keystrokes or mouse clicks.
- **Misrepresentation**. This involves the actual deception of users and system operators.

Hardware misuse:

- **Logical scavenging**. This involves scavenging through discarded media.
- **Eavesdropping**. This involves intercepting communications.
- **Interference**. This involves the actual interference of communications, such as in jamming communications, or modifying it in some way.
- **Physical attacks**. This involves an actual physical attack on the hardware.
- **Physical removal**. This involves the actual physical removal of hardware.

Masquerading/spoofing:

- **Impersonation**. This involves the impersonation of a user/device.
- **Piggy back attacks**. This involves adding data onto valid data packets.
- **Spoofing**. This involves the spoofing of devices.
- **Network weaving**. This involves confusing the system on the where-abouts of a device, or confusing the routing of data.

Pests:

- **Trojan horses**. This involves users running programs which look valid, but install an illicit program which will typically do damage to the host.
- **Logic bombs**. This involves the installation of a program which will trigger at some time in the future based at a given time or event.
- **Malevolent worms**. This involves a worm program which mutates in a given way which will eventually reduce the quality-of-service of the network system, such as using up CPU resources, or taking up network bandwidth.
- **Viruses**. This involves attaching program which self replicate themselves.

Bypasses:

- **Trap door impersonation**. This involves the creation of pages or login screens which look valid, but are used to gain information from a user, such as for their bank details, or login password.
- **Authorization attacks**. This involves trying to gain access to a higher level of authorization than is valid for the user, such as with password attacks.

Active misuse:

- **Active attack**. This is the entering incorrect data with the intention to do damage to the system.

- **Incremental attack**. This involves damaging a system using an incremental approach.
- **Denial-of-service**. This involves attacking a host with continual requests for services, which eventually reduces its performance.

Passive misuse:

- **Browsing**. This issues random and/or selective searches for information.
- **Interference/aggression**. This involves exploiting database weaknesses using inferences.
- **Covert channels**. This involves hiding data in valid network traffic.

1.14 Defence-in-depth

Another term which borrows from military activities is defence-in-depth, which aims to put as many lines of defence in the face of an intruder in order to slow them down. With this it is then easier to detect their movements before they can do any damage, as illustrated in Figure 1.14. This strategy normally requires some form of intrusion detection between the levels of penetration into the system. Thus, the more levels of defence, the longer it is likely to take for an intruder to gain an advantage on a system. For example, in Figure 1.16, there are many obstacles, such as firewalls, placed in the way of the intruder. In this case the intruder must gain entry to the main gateway, which is often relatively easy, and then transverse into the network over each of the firewalls, each of which perform a more in-depth check on the validity of the data packets. The figure also contains intrusion detection agents and system logging programs which detect intrusion. A good security strategy is thus to refine the security levels, through the levels of defence, as it normally gets more difficult to progress through these levels, as they get nearer the focus of an intrusion. Along with this, the different levels should provide a different challenge for each level, as it does little good to face the same challenge for different levels. For example, the first level could check for the destination and source addresses, while the next level could examine the source and destination TCP ports, and then the next level checks for a user name and a password, while the last level could require some form of electronic certificate which verifies the user.

One of the weakest areas of security over the past decade has been in wireless systems. There are three main reasons for this: the lack of defined standards; the weakness of many of the initial security standards; and the relative openness of wireless systems. In WEP, wireless systems were easily crackable in a relavitely short time, and once the key was cracked, it could then be used to crack the rest of the communications (as the encryption key was shared around the hosts on the wireless network). The openness of wireless, and its weak security protocols have thus allowed the development of robust security protocols which are now being applied into general system security, and are providing for a framework which tries to verify not only hardware systems, but also users. This level of authentication is obviously a key element in any defence-in-depth strategy, as the system must guard against

spoof attacks, which is similar to a spy in a military environment, who may disguise their homeland and ID with fake travel documents and a fake passport.

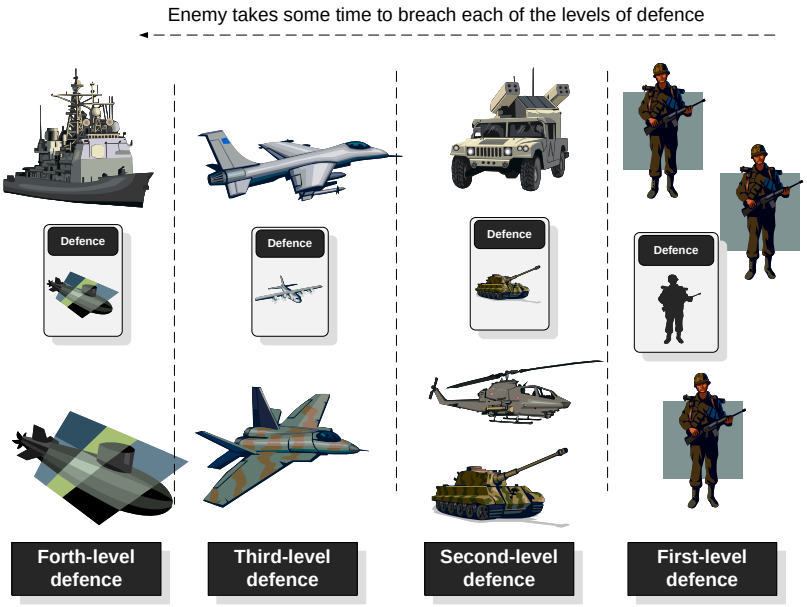

Figure 1.14 Defence-in-depth

1.15 Gateways and DMZ (Demilitarized Zones)

There are many similarities between military operations and network/host security. In a military situation we often define: a trusted zone: an untrusted zone: and a demilitarized zone (DMZ). In a war situation, it is in the DMZ that trusted and untrusted troops can mix, as illustrated in Figure 1.13. In terms of network traffic a firewall device is used to filter trusted and untrusted network traffic. Normally the trusted side is named **inside**, and the external side as **outside**. In this way the servers, which can be accessed from outside the network, are placed in the DMZ, so that it is not possible for untrusted traffic to enter the main internal network, as illustrated in Figure 1.15.

The zones, such as inside, outside, and DMZ, can then be classified with their security level, such as the inside network having the highest security level, the DMZ the next highest, and outside with the lowest level. By default traffic is often trusted to go from a higher security level to a lower one (from inside to DMZ, and from the DMZ to the outside), but not trusted to go from a lower level to a higher one. Thus level of trust, though, assumes that attacks come from the outside zone, but many threats originate from the inside zone. Along with this we have the concept of the incoming traffic, which is the traffic flowing from a lower zone to a higher zone, and, thus, the outgoing traffic flows from a higher security zone to a lower one.

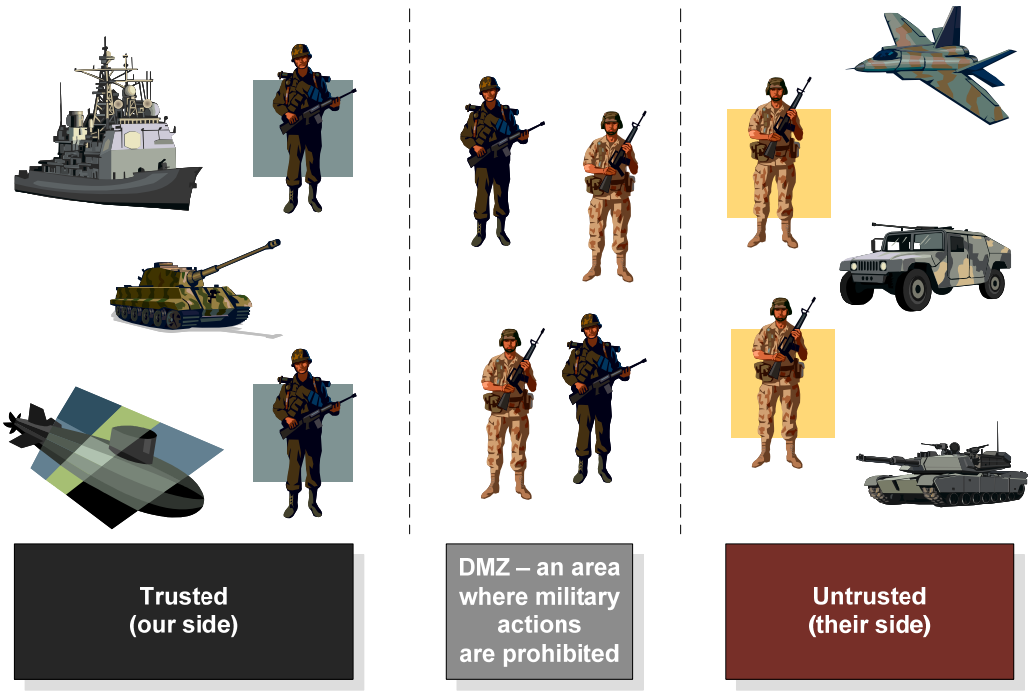

Figure 1.15 DMZ

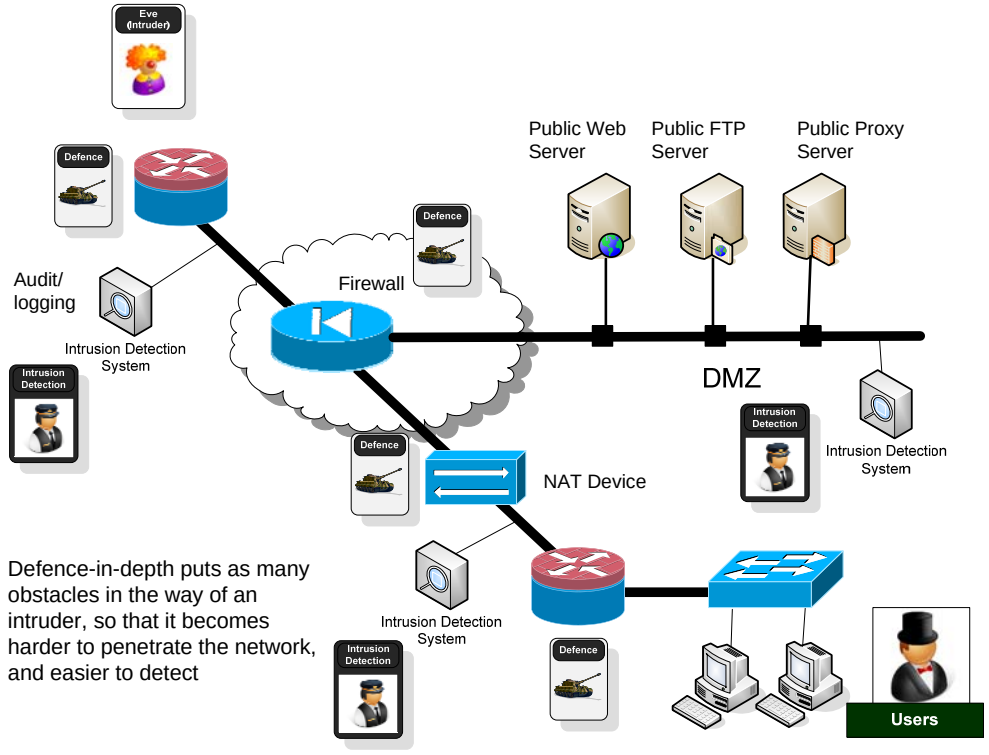

Figure 1.16 DMZ

1.16 Layered model and security

Security is an extremely complex issue, and it would be almost impossible to secure every bit of data, without splitting the task of security into different layer of abstration. With this approach the overall system is split into different entities, each of

which are analysed for their security weaknesses, and, if possible, a defence strategy is implemented to overcome them. The overall system will hopefully be secure if all the sub-elements are correctly secured. Unfortunately, though, many systems fall-down in that not all of the sub-elements are properly secure, and can thus let the overall system down, as they can provide a means of intrusion into the system.

Often the OSI (Open Systems Interconnection) model is used for network communication, where the transmission and reception of the data is split into key functions, each of which has a defined objective. Unfortunately, each layer can have information that can be used by an intruder, such as network source and destination IP addresses in the network layer, or the TCP ports used in the transport layer. An important factor is that the data and protocol information can be best protected with encryption. It is thus important to know the information which must be protected, and to protect it against an intruder stealing it, or even changing it. Figure 1.17 gives examples of some of the protection methods used at different levels in order to protect the data transmission. If the data itself must be protected it is normally encrypted before it is transmitted (or stored). Typical encryption methods include RSA and DES.

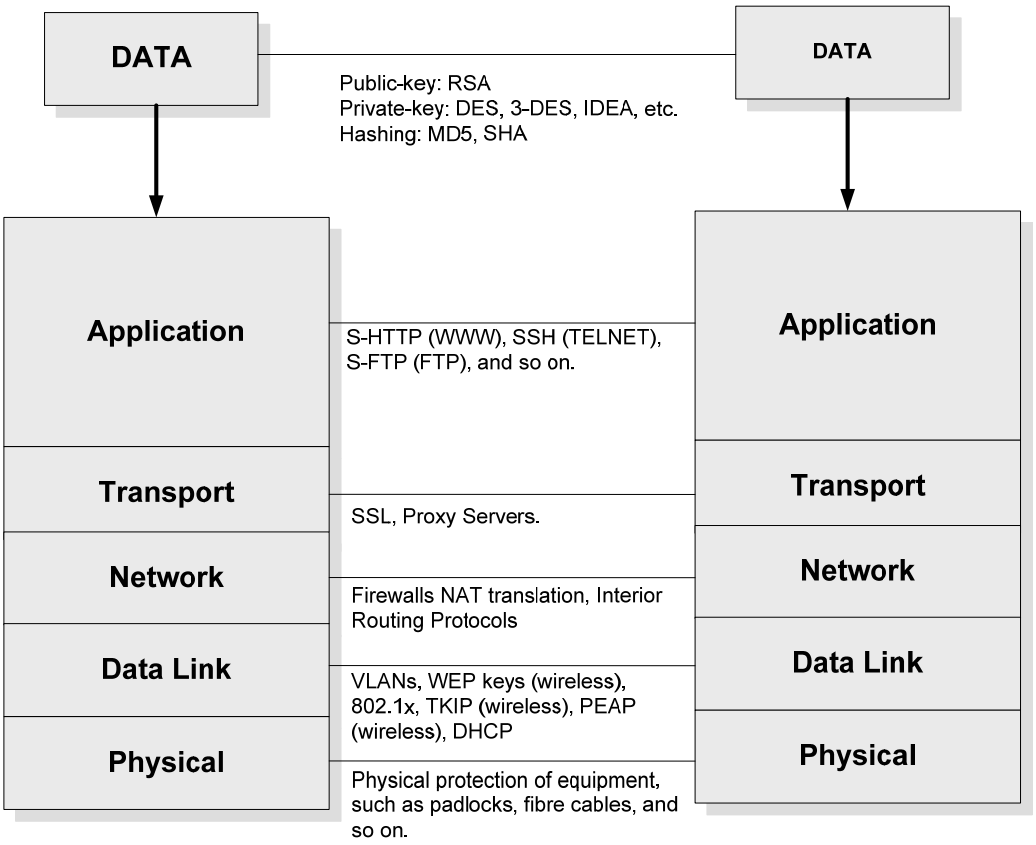

Figure 1.17 OSI model and security

Some of the methods used at each layer include:

- **Application Layer**. At the application layer, the application layer protocols, such as FTP, TELNET and HTTP, are typically not secure and often send their information, including passwords and user ID's, in a plain text form. A great improvement are the secure protocols, such as S-FTP (which is a secure replacement for FTP), S-HTTP (which is a secure replacement for HTTP), and SSH (which is a secure replacement for TELNET).
- **Transport Layer**. Below the application layer is the transport layer, which is responsible for creating a reliable connection between two hosts. At this layer, SSL (secure sockets layer) and proxy servers can be used to hide information about a connection.
- **Network Layer**. At the network layer the main information to be hidden is the source and destination network addresses, and encryption methods, such as IP-Sec can be used to encryption the contents of the data packet. Along with this NAT (Network Address Translation) can be used to hide network addresses, for both the source and/or the destination.
- **Data Link Layer**. Below the network layer, at the data link layer, a VLAN (Virtual LAN) can be used to segment networks, so that hosts cannot communicate with each other, unless they are allowed to. In a wireless network, the data frame transmitted can be encrypted with WEP (Wireless Equivalent Privacy) or TKIP (Temporal Key Integrity Protocol) or with more complex encryption, such as AES (Advanced Encryption Scheme).
- **Physical Layer**. At the lowest layer, the physical layer, the network requires to be protected against damage to cables, hardware, closets, and so on. The methods at this level are padlocks, cable ducts, RF shielding, and so on.

1.17 Encryption and a layered approach to defence

Encryption normally involves converting plain text into cipher text, with a known algorithm, such as RSA or DES, and a unique electronic key (this will be explained in more detail in Unit 3). A key approach to defending a network and its hosts is to develop a layered approach to defence. With the OSI model, information can be gained at each level, thus it is important to know what requires to be protected. Examples information that can be contained within each layer is:

- **Physical**. Types of cables used, location of hosts/servers, power supplies, and so on.
- **Data link**. MAC/physical addresses, source and destination MAC addresses.
- **Network**. Source and destination network addresses (such as IP address).
- **Transport**. Source and destination ports (such as TCP/UDP ports).
- **Session/Presentation/Application**. This can contain session and user information, such as the pages that are accessed on a Web server, and a user's ID and password.

One of the best methods of protecting data is to encrypt it. This can be done, if necessary, at each layer of the OSI model. So, if a layer is encrypted from the top level, then the data held within the layer, and all upper layers will be protected. For example, if the transport layer is encrypted then the data in the application layer will also be encrypted. In certain cases the encryption can be achieved for a lower layer, such as in wireless networking, which encrypts at the network layer. In this case, when the data packets go up to a layer above the network layer, the data packets are unencrypted. This is the case for wireless network with encryption, where the transmission across the network medium, which is free-space, and once received, it is decrypted back into normal data packets.

1.18 Software Tutorial – Data packet capture

Visual Studio will be used in this module. An important library to use is WinPcap [2], which allows for the capture of data packets. Once the download is complete, implement the following:

1.18.1 Download the latest version of WinPcap [3], and once installed, download the solution [1]:

🕮 **Web link:** http://buchananweb.co.uk/srcSecurity/unit01_1.zip

Demo: http://buchananweb.co.uk /media/unit01_1_demo.htm

which has the following code [1]:

```
using System;
using Tamir.IPLib;

namespace NapierCapture
{
   public class ShowDevices
   {

      public static void Main(string[] args)
      {
         string verWinPCap =null;
         int count=0;

         verWinPCap= Tamir.IPLib.Version.GetVersionString();

         PcapDeviceList getNetConnections = SharpPcap.GetAllDevices();

         Console.WriteLine("WinPCap Version: {0}", verWinPCap);

         Console.WriteLine("Connected devices:\r\n");

         foreach(PcapDevice net in getNetConnections)
         {
            Console.WriteLine("{0}) {1}",count,net.PcapDescription);
            Console.WriteLine("\tName:\t{0}",net.PcapName);
            Console.WriteLine("\tMode:\t\t\t{0}",net.PcapMode);
            Console.WriteLine("\tIP Address: \t\t{0}",net.PcapIpAddress);
            Console.WriteLine("\tLoopback: \t\t{0}",net.PcapLoopback);

            Console.WriteLine();
            count++;
         }
         Console.Write("Press any <RETURN> to exit");
         Console.Read();
      }
   }
}
```

Run the program, and verify that it produces a list of the available network cards, such as:

```
WinPCap Version: 1.0.2.0
Connected devices:

0) Realtek RTL8169/8110 Family Gigabit Ethernet NIC
    (Microsoft's Packet Scheduler)
        Name:   \Device\NPF_{A22E93C1-A78D-4AFE-AD2B-517889CE42D7}
        Mode:                   Capture
        IP Address:             192.168.2.1
        Loopback:               False

1) Intel(R) PRO/Wireless 2200BG Network Connection (Microsoft's Packet Scheduler)
        Name:   \Device\NPF_{044B069D-B90A-4597-B99E-A68C422D5FE3}
        Mode:                   Capture
        IP Address:             192.168.1.101
        Loopback:               False
```

1.18.2 Next update the code so that it displays the information on the network connections [1]:

```
        foreach(PcapDevice net in getNetConnections)
        {
           Console.WriteLine("{0}) {1}",count,net.PcapDescription);

           NetworkDevice netConn = (NetworkDevice)net;

           Console.WriteLine("\tIP Address:\t\t{0}",netConn.IpAddress);
           Console.WriteLine("\tSubnet Mask:\t\t{0}",netConn.SubnetMask);
           Console.WriteLine("\tMAC Address:\t\t{0}",netConn.MacAddress);
           Console.WriteLine("\tDefault Gateway:\t{0}",netConn.DefaultGateway);
           Console.WriteLine("\tPrimary WINS:\t\t{0}",netConn.WinsServerPrimary);
           Console.WriteLine("\tSecondary
WINS:\t\t{0}",netConn.WinsServerSecondary);
           Console.WriteLine("\tDHCP Enabled:\t\t{0}",netConn.DhcpEnabled);
           Console.WriteLine("\tDHCP Server:\t\t{0}",netConn.DhcpServer);
           Console.WriteLine("\tDHCP Lease Ob-
tained:\t{0}",netConn.DhcpLeaseObtained);
           Console.WriteLine("\tDHCP Lease Ex-
pires:\t{0}",netConn.DhcpLeaseExpires);
           Console.WriteLine();
           count++;
        }
```

A sample run shows the details of the network connections [1]:

```
1) Intel(R) PRO/Wireless 2200BG Network Connection (Microsoft's Packet Scheduler)
        IP Address:             192.168.1.101
        Subnet Mask:            255.255.255.0
        MAC Address:            0015003402F0
        Default Gateway:        192.168.1.1
        Primary WINS:           0.0.0.0
        Secondary WINS:         0.0.0.0
        DHCP Enabled:           True
        DHCP Server:            192.168.1.1
        DHCP Lease Obtained:    03/01/2006 10:44:40
        DHCP Lease Expires:     04/01/2006 10:44:40
```

List the details of the connections on your PC:

1.18.3 Update the code from 1.18.1 with the following code [1]. In this case the 2nd connection is used (getNetConnections[1]) in a promiscuous mode (change, as required, depending on your local network connection).

```
using System;
using Tamir.IPLib;
using Tamir.IPLib.Packets;

namespace NapierCapture
{
   public class CapturePackets
   {
      public static void Main(string[] args)
      {
         PcapDeviceList getNetConnections = SharpPcap.GetAllDevices();

         // network connection 1 (change as required)
         NetworkDevice netConn = (NetworkDevice)getNetConnections[1];
         PcapDevice device = netConn;

         // Define packet handler
         device.PcapOnPacketArrival +=
            new SharpPcap.PacketArrivalEvent(device_PcapOnPacketArrival);

         //Open the device for capturing
         //true -- means promiscuous mode
         //1000 -- means a read wait of 1000ms
         device.PcapOpen(true, 1000);

         Console.WriteLine("Network connection: {0}", device.PcapDescription);

         //Start the capturing process
         device.PcapStartCapture();

         Console.Write("Press any <RETURN> to exit");
         Console.Read();

         device.PcapStopCapture();
         device.PcapClose();
      }
      private static void device_PcapOnPacketArrival(object sender, Packet packet)
      {
         DateTime time = packet.PcapHeader.Date;
         int len = packet.PcapHeader.PacketLength;
         Console.WriteLine("{0}:{1}:{2},{3} Len={4}",time.Hour, time.Minute,
                   time.Second, time.Millisecond, len);
      }
   }
}
```

Run the program, and produce some network traffic and verify that it is capturing packets, such as:

```
13:17:56,990 Len=695
13:17:57,66 Len=288
13:17:57,68 Len=694
13:18:4,363 Len=319
13:18:4,364 Len=373
13:18:4,364 Len=371
13:18:4,365 Len=375
13:18:4,366 Len=367
```

1.18.4 Update the code with a filter. In the following case an **IP** and **TCP** filter is used [1]:

```
         device.PcapOpen(true, 1000);
```

```
        Console.WriteLine("Network connection: {0}", device.PcapDescription);

        string filter = "ip and tcp";

        //Associate the filter with this capture
        device.PcapSetFilter( filter );

        //Start the capturing process
        device.PcapStartCapture();
```

Generate some data traffic, such as loading a Web page, and show that the program is capturing the data packets.

1.18.5 Next update the filter so that it only captures **ICMP** packets, such as:

```
string filter = "icmp";
```

Generate some data traffic, and prove that it does not capture the packets. Now ping a node on your network, such as:

```
Ping 192.168.1.102
```

And prove that it captures the data packets, such as:

```
13:40:47,761 Len=74
13:40:48,756 Len=74
13:40:48,759 Len=74
13:40:49,757 Len=74
13:40:49,760 Len=74
13:40:50,757 Len=74
13:40:50,760 Len=74
```

The source code for this is at:

🕮 **Web link:** http:// buchananweb.co.uk/srcSecurity/unit01_2.zip

🕮 **Web link:** http:// buchananweb.co.uk/media/unit01_2_demo.htm

1.18.6 Investigate each of the following, one at a time, capture filters and determine the operation:

```
string filter = "not arp";
string filter = "port 80";
string filter = "host www.google.com";
string filter = "(port 80) and (host www.intel.com)";
string filter = "port 53";
```

1.18.7 Using IPCONFIG /ALL, determine the MAC address of your main adapter, and use an Ethernet filter with your MAC address, such as:

```
string filter = "ether host 00:79:7e:cc:c8:b7”
```

1.18.8 Install Wireshark or Ethereal, using TSHARK (the command line version of Wireshark) or TETHEREAL (the command line version of Ethereal), apply some of the filters from Tutorial 1.18.6, such as:

```
tshark -V -i 3 -f "ether host 00:79:7e:cc:c8:b7" -Sw 1.out -T text
```

where "-i 3" represents the forth interface and 1.txt represents the output text file.

1.19 Chapter Lecture and Exercises

View the lecture at:

http://www.asecuritysite.com/security/information/chapter01

1.20 References

[1] This code is based on the code wrapper for WinPCap developed by T.Gal [http://www.thecodeproject.com/csharp/sharppcap.asp].

[2] http://www.winpcap.org/

[3] http://www.winpcap.org/ install/default.htm