Université d'Ottawa · University of Ottawa

Université d'Ottawa · University of Ottawa

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

Ali AABAS

AUTEUR DE LA THÈSE - AUTHOR OF THESIS

M. Sc. (Systems Science)

GRADE - DEGREE

Systems Science Program

FACULTÉ, ÉCOLE, DÉPARTEMENT - FACULTY, SCHOOL, DEPARTMENT

TITRE DE LA THÈSE - TITLE OF THE THESIS

Analysis and Design of an Internet's Security Taxonomy

A. El Saddik

DIRECTEUR DE LA THÈSE - THESIS SUPERVISOR

A. Miri

CO-DIRECTEUR DE LA THÈSE - THESIS CO-SUPERVISOR

EXAMINATEURS DE LA THÈSE - THESIS EXAMINERS

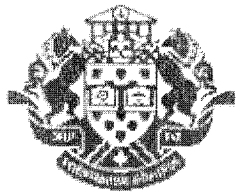
S. Shirmohammadi

J. Zhao

J-M. De Koninck, Ph D

LE DOYEN DE LA FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

DEAN OF THE FACULTY OF GRADUATE AND POSTODORAL STUDIES



University of Ottawa
Systems Science

Analysis and Design of an Internet's Security Taxonomy

By

Ali Abbas

A thesis submitted to
The Faculty of Graduate and Postgraduate Studies

In Partial Fulfillment of the degree of
Master of Systems Science

School of Information Technology and Engineering
Systems Science
University of Ottawa

Your file Votre référence
ISBN: 0-494-01392-3
Our file Notre référence
ISBN: 0-494-01392-3

To my wife Sausan;
To my kids, Al-Hassan, Tabarek, and Amir;
Those from whom the time was stolen,
To make this work possible

Acknowledgments

The accomplishment of this work is greatly due to the huge encouragement, unconditional support and wonderful guidance that I have received from my supervisors, Dr. Abdulmotaleb El Saddik and Dr. Ali Miri. Therefore, I would like to offer my greatest thanks to both of them for their generous giving of their time and expertise. I would also like to thank my colleague Ismail Shakra for his help in proofreading and support with the language and also for his encouragement.

Last but certainly not least, I must extend a special thanks to my family; wife and kids, whose amazing support and ability to cope with the difficulties have been vital in such a seemingly endless journey of studies and research.

Table of Contents

List of Figures

List of Tables

Abstract

The main objective of the different security services and mechanisms today are to provide privacy of information and to ensure that the tools used to establish a proper environment to the user are reliable and trusted. To provide a high level of security, an exhaustive approach that depicts a complete package of the Internet security classification categories is needed to show how, where and when each of the security services works. The proposed comprehensive Internet security taxonomy is innovative in its overall study of Internet security as the guidelines established to assess the taxonomy are strict. Many of the known Internet security attacks and the Internet security services associated with them are analyzed in this proposed taxonomy. A mapping of the security services against the security attacks and corresponding countermeasures is given. An assessment of the performance of the proposed taxonomy is also given, showing it to be useful, exhaustive and unambiguous.

CHAPTER I

Introduction

The explosive growth in computer systems and their interconnections via networks and the Internet has increased the dependence of big organizations and individuals on the information stored and communicated using these systems. This has led to a heightened awareness of the need to protect data and resources from disclosure, to guarantee the authenticity of data and messages, and to protect systems from network-based attacks. Cryptography and diverse network security approaches have matured, leading to the development of practical, readily available applications to enforce Internet security.

The complexity of security attacks has rendered countersecurity measures just as complex such that they have been evolving by employing different methods and algorithms. The first assumption about the security of a computer and network system is that it is intact if the computer and the network behave as expected. Moreover, computer and network systems are considered even more secure if the software that controls the operation of the computer and the data communication behaves as expected. However, most systems do not behave as such. As a result, securing the network systems has become an important issue, and the notion of computer security, network security, and the Internet security is attracting more and more research.

Although Internet security, and computer and networks security share many common issues, there are some open questions that are unique to Internet security discipline. One of the main reasons behind that is due to the great possibility of unexpected behaviour that can occur over the Internet. In a simplified description, the Internet performs as a two-way network, and it has a huge publication capacity. As the Internet has both, the potential and the ability to publish information and services to millions of users, it also provides a suitable environment for hackers, crackers, criminals, vandals, and others to break into computer systems. Breaking into computer systems has

many different goals and objectives, presenting essential risks to the Internet environment, and posing weaknesses or limitations that need to be tackled.

Recently, the Internet has become the fastest growing part of the networks applications; most importantly, it is the part that attracts more security attention because of its vulnerability to the attacks. A successful attack on an Internet system presents a real problem because the attack can be noticed by millions of people within a very short time. Each attack has a reason behind it; for example, an attack can simply be a random act of vandalism, or it may have a financial, technical or even ideological reason.

The web, which is one of the Internet applications, is by itself, easy to use but the web applications and browsers are complicated software that may encounter many security challenges and may have security flaws. In previous times, many features have been implemented within the web servers or browsers without paying careful attention to the security requirements. One important feature of the web server has been that of an extensible design. Unfortunately, the web server's extensibility makes it vulnerable, if not implemented properly. For example, the web browser may connect with a database, web-based applications, or programs running on any system's network, etc. If not properly implemented, modules such as CGI or php that are added to a web server can compromise the security of the entire system. The same principle that applies to the web server applies as well to the web client, which can also be extensible through scripting capabilities or plug-Ins, etc.

Another major issue is the dependability of many Internet applications on the TCP/IP protocol, which has security limitations. For example, the TCP/IP protocol is subject to disruption of service through denial of service attacks. As a result, an awareness of the vulnerabilities and security flaws of the technology being used is essential. For example, in case of the TCP/IP one must take precautions against a major service disruption and must, at the same time, prepare the security system with countermeasures to the denial of service attacks.

The Internet security flaws and vulnerabilities as well as the wide range of technologies and techniques used to implement and utilize different Internet applications emphasize the complexity of connecting all those related issues together and mapping them through classification categories to produce taxonomy. Such taxonomy is the key to understanding the security threats that the Internet is facing today, and the countermeasure approaches that should be devised in order to keep the Internet secure.

1.1 What is Taxonomy and why is it needed?

Taxonomy is a method of classifying general scientific principles. Such classification includes events, items, aspects, etc. "Taxonomy (from Greek *taxis* meaning arrangement or division and *nomos* meaning law) is the science of classification according to a pre-determined system, with the resulting catalogue used to provide a conceptual framework for discussion, analysis, or information retrieval. In theory, the development of a good taxonomy takes into account the importance of separating elements of a group (taxon) into subgroups (taxa) that are mutually exclusive, unambiguous, and taken together, including all possibilities. In practice, a good taxonomy should be simple, easy to remember, and easy to use" [Taxonomy, 2003]. In general, classification allows us to map all such items in order to systematically deal with them. This research presents a classification of the actual security services and attacks, and thus provides a guide to Internet system designers, programmers, analysts, administrators, and users. It also allows for the identification of areas of research that are still relatively unexplored.

The first and the most challenging step in writing an Internet security taxonomy is to find practical, efficient, and exhaustive classification categories. A good approach toward taxonomy should have classification categories with the following characteristics [Amoroso, 94]:

1) **Mutually exclusive** - should be such that the result categories do not have common characteristics,
2) **Exhaustive** – should be taken together, the categories should cover all potentials,
3) **Unambiguous** – should be obvious and specific so that a categorization is not uncertain, regardless of what is being classified,
4) **Repeatable** – the category should be applicable to a diverse kind of applications.
5) **Accepted** – should be rational and realistic so that it could become generally accepted and standard,
6) **Useful** – it can be expanded and can be utilized to provide an insight into the field of inquiry.

All or some of those characteristics can be used to gauge the efficiency of any possible taxonomy. It is understandable that these characteristics should be available for any taxonomy in order to provide a complete approach and a useful result. By maintaining these characteristics, the proposed taxonomy will provide a good understanding in its field of study: the Internet security. Moreover, in order for our taxonomy to provide the kind of understanding the researchers are looking for, it has to be measured and assessed against the above described characteristics. This will produce a considerable challenge as the Internet security in a fast moving technology.

1.2 The Motivations and Contributions of this Thesis

To the best of our knowledge, there has not been a complete study that classifies and categorizes the Internet security. There are many proposed security taxonomies but they are either partially done due to the effort required to produce an exhaustive taxonomy, or have been retracted to specific applications. With the dramatic increase of the use of the Internet and its applications that require high level of security services, such as E-Commerce transactions and on-line banking, it is quiet useful to formulate a systematic approach to analyze the security services as countermeasures to directly associated security attacks.

The research aspect of this thesis has been motivated by the desire to answer many essential questions regarding Internet security services. The Internet security

measures require a deep understanding of the methods of security attacks, in order to provide the essential tools to develop a security defence and evolve the overall security outcomes. Our research can be used as a reference or a handbook for helping the software engineers and scientists who are interested in the field of Internet security services.

This thesis can be seen as an attempt to provide an exhaustive, yet practical approach to formalize classification categories that map the security attacks to the security services associated with the attacks. We will perform an analysis and assessment of the current algorithms being used in today's security services. This analysis will provide an evaluation of the risks and benefits of each one of those algorithms, and will be used to present an extensive analysis and classification of the most used Internet security algorithms, the security attacks on the systems, and the security countermeasures developed to deal with these attacks. Below is a list of the contributions of this thesis:

- Developed new classification categories for the Internet security attacks based on the threats and potential threats in today's technologies.
- Proposed a new Internet security taxonomy that mapped the security services with the classes of the security attacks that we developed.
- Identified the security attacks within each class of category.
- Identified existing solutions and countermeasures to the above identified attacks.
- Associated each security attack with the appropriate security service which is being violated by that attack.
- Assessed the performance of the proposed taxonomy with respect to the taxonomy's characteristics.
- Provided a comprehensive survey of exiting and related taxonomy.

1.3 Thesis Outline

Chapter 1 introduces an overview of Internet security, objectives, thesis contributions and motivations. Chapter 2 discusses the necessary background of basic security related issues and dissections such as: security services, security attacks, models, security mechanisms, techniques (e.g. Cryptography), and biometrics authentication process. Chapter 3 analyzes the relevant works related to security taxonomy, explaining their strengths and weaknesses, evaluating some of their performance and drawing a comparison between them. In Chapter 4 the focus is on the development of our new classification categories and new Internet security taxonomy of the types of security services and attacks. Categorizing actual and possible Internet security attacks will be introduced. The attacks within each category and existing solutions that deal with the attacks will be identified. Chapter 5 concludes this research by highlighting the achievements accomplished and by anticipating future work.

CHAPTER II

Internet Security Service's Technologies

In the early days of the Internet, the focus was on the availability of communication services and their functionalities. The need for security only came to pass after it became obvious that the Internet cannot operate properly without security services that can stop or reduce the risks of vulnerability attacks. Security attacks and countermeasures have evolved over the years by hackers and system administrators monitoring each other's techniques and algorithms. Security on the Internet has to involve some measures to ensure the security of information and to gain people confidence in order to enhance the use of the Internet. "The web's highest goal was seamless availability. Today, with an internationally connected user network and rapidly expanding Web functionality, reliability and security are critical. Vendors engaged in retrofitting security must contend with the Web environment's peculiarities, code and user mobility, and stranger-to-stranger communication" [Rubin, 1998].

An important question to ask about any security measure is whether or not it is truly necessary in a given situation. Although this is a question that often hinges on technical questions, it also depends on questions of organizational policy and the motivation behind its security activities; hence it depends on the level of security that each individual or organization desires. Accordingly, we can assess the level of security which different organizations needs, why they need it and what are the objectives behind operating specific security services [Erik, 2002]. Therefore, we can develop security services that meet the objectives of any particular need. To be more specific, different applications have different security considerations. Many different parties communicate with each other using today's Internet and in most cases they do not know enough about each other. This fact should be taken into consideration and extra cautious should be given when research about the Internet security is performed.

2.1 Security Attacks

Security attacks are actions that exploit the vulnerabilities of a given system to gain unauthorized access and/or unauthorized use. There are many different ways and methods to defeat security systems. A simple and typical offensive step against security systems is to apply a trial and error technique to break into the system by trying to take over the password file of the system. From the simplest to the most complicated one, security attacks can be classified according to the nature of attacks, the tools they use, and the objectives of the attack [Richard, 1997].

The first security measure against such type of attacks is to keep an audit trail of attempts to log on to the system. An audit trail is a record of significant events within the system and is a common feature in modern computing systems. A log file can be used to show if a user account is a target of an attack [Garfinkel, 2002]. For good discussion on the attacks mechanisms and attacks detection we refer to Lukatsky [Lukatsky, 2003].

Internet security attacks are getting more automated, "The level of automation in attack tools continues to increase. Automated attacks commonly involve four phases, each of which is changing. These phases include scanning for potential victims, compromising vulnerable systems, propagating the attack, and coordinated management of attack tools" [Householder, 2002]. Just as the attacks technologies are getting smarter, so are the security countermeasures, which are getting more sophisticated and evolved. This has led to the introduction of many security services at different levels of the communication protocols. For example, there are among others, security measures at the Network level (IPSEC) [Carlton, 2001] [Niels, 1998], at the transaction level (SSL) [Introduction SSL] [SSL], and at the application level (SHTTP) [Adam, 1995].

Identifying security attacks as passive or active is a useful mean and wide framework of classifying them. A passive attack attempts to learn or make use of information from the system but does not affect the system resources. Passive attacks by nature are a simple monitoring of information flow. The most common type of passive attack is the observation of messages' content; another common type is traffic analysis.

There are some other forms of passive attacks that may reveal some of users' privacy with regard to their activities on the Internet. For example, the list of Web locations visited by a user often conveys detailed information concerning the user's interests, financial information or health records. As a result, users need to consider their Web-browsing history to be private information that they do not want unknown parties to get access to. In fact, passive attackers can get advantage of such a history record to analyze the information and abuse it. This is why users have to be guaranteed some assurance that information regarding the visit to any specific site is not available to other parties [Edward, 2000].

Active attacks, on the other hand, attempt to alter system resources or affect their operation. These attacks can target any entity within the Internet communication model. They may involve both modification and creation of false data streams and can be subdivided into four categories: masquerade, replay, modification of message, and denial of service. In general, the security attacks can occur everywhere within the Internet communication model as depicted in Figure 1. Most common attacks in this category can be divided in to the following [Dan, 2000]:

- Physical Attacks; such as damaging the hardware.
- Impersonation Attacks; such as stealing the identity and masquerading as another person.
- Integrity Attacks; such as deleting, adding, or modifying data
- Disclosure Attacks; such as revealing private data.
- Denial of service Attacks; such as preventing authorized user form accessing the system or data.

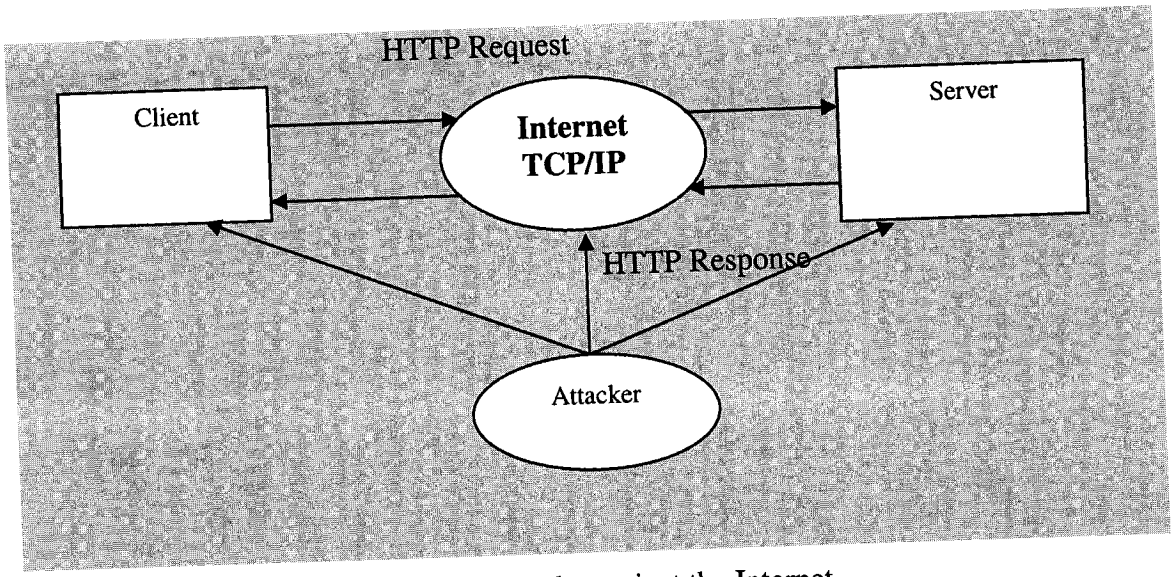


Figure1: Attacks against the Internet

A framework for studying most common security threats was suggested by John D. Howard and Thomas A. Longstaff [Longstaff, 1998]. According to their specification the security attacks are series of steps or events that can occur on a computer network, which result in an action that is not authorized to happen. In general, a security attack consists of several elements. The first element is the attempt of the attacker to gain unauthorized access into the system by using specific tools to exploit any vulnerability and/or accessibility to the target. The second element is the unauthorized action that results because of the access that the attacker has gained. The third and last element is the consequence or the damage that may come about (see Figure 2).

The first two steps (tools and vulnerability) in a series of steps to accomplish an attack are used to assess the kind of action that might be taken against the target. In other words, the attackers will use specific tools to exploit a vulnerability of the target to conclude the best action that may give a required result. Eventually, and in order for the attack to be considered successful, the last step is to gain an unauthorized result. The crucial point is to have a clear cut to differentiate between what is authorized and what is unauthorized, this differentiation is important because an authorized result is not an attack.

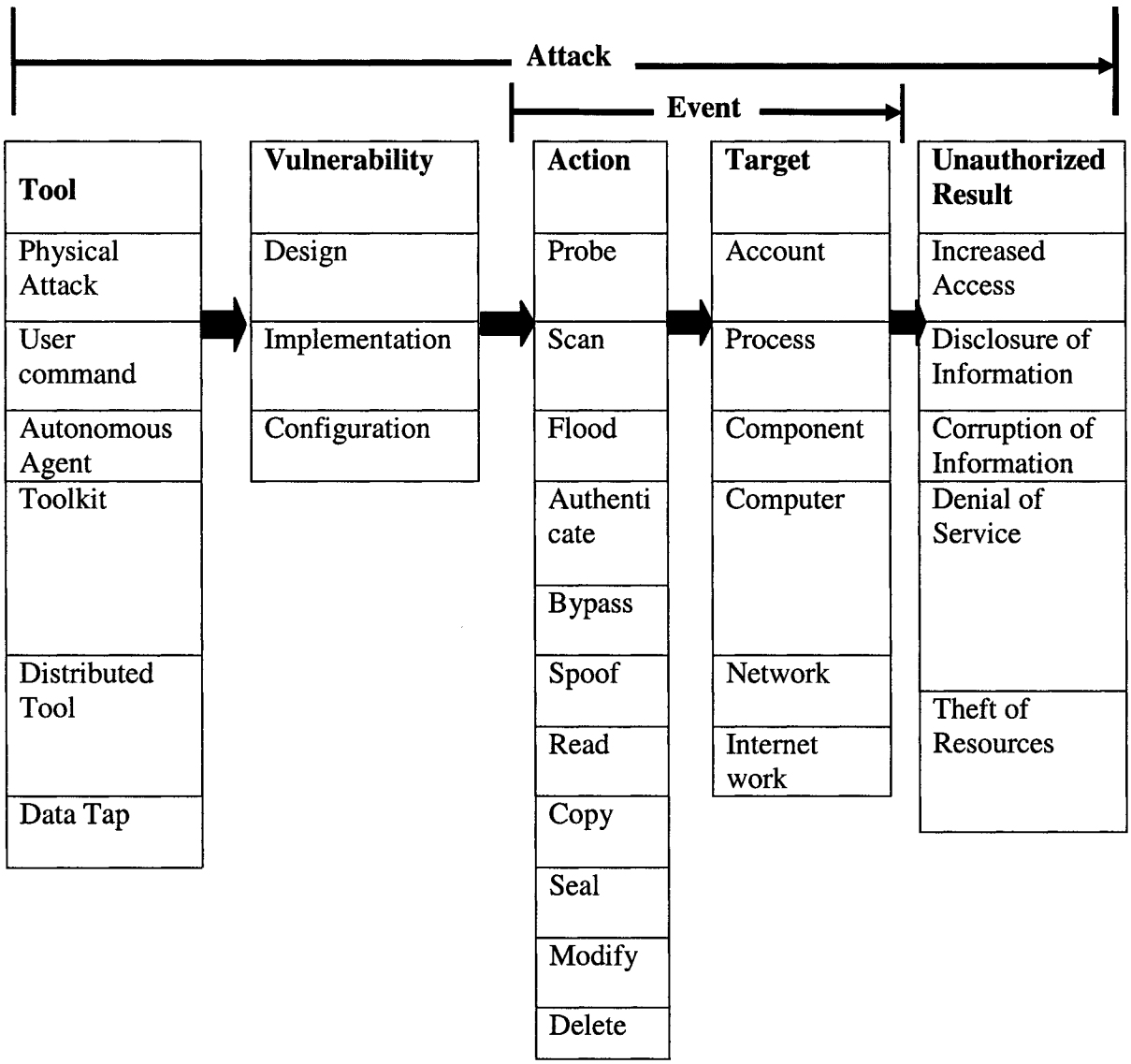


Figure 2: Attacks Model of Longstaff

2.2 Traditional Security Services

Security services provide protection for the transmitted or stored data against any threats or attacks. Securing such data in any Internet application will require special considerations that come in conjunction with the nature of the application itself. Internet security flaws may result from the vulnerabilities embedded within the code of any Internet utilization such as any Web based application [Scott, 2003] [Yao-Wen, 2003]. In order to assess the security needs of any Internet system effectively, and to evaluate

and choose various security products and policies, we need to understand the nature of the vulnerabilities, and the reasons behind securing specific information.

Data security services provide a sense of security which is necessary to gain people's confidence when they store confidential and sensitive data or when they transmit such data. Much of the activity of humankind, in areas such as e-commerce, foreign policy, military action, and personal interactions, depends on the use of documents and the different parties that are sharing or exchanging these documents. Documents typically have signatures and dates. They may need protection from disclosure, tampering, or destruction; they may be notarized or witnessed; recorded or licensed, etc. The security service itself comes as a combination of three different levels or security mechanisms which build the hierarchical structure of the security services and can provide the required security countermeasures. These are: Cryptography, Authentication, and Communication security [Shweta, 2002], Figure 3 shows the hierarchical structure of the web security.

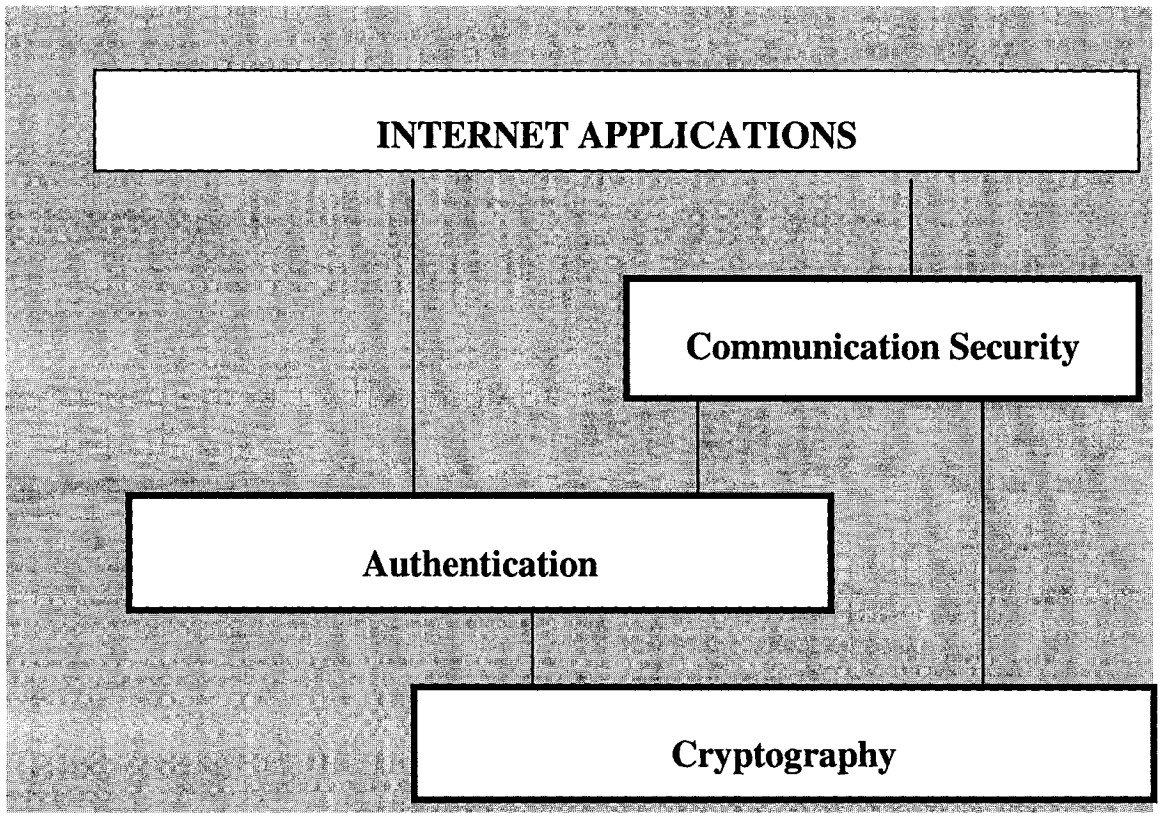


Figure 3: Security Service Hierarchy

2.2.1 Security Mechanisms

A security mechanism is a process or method that protects a security service against attacks that might happen within the system. In fact there is no mechanism that can achieve all of the required security services together, such as, detection, prevention and/or recovery. However, there is a specific technique that is a corner stone for most of the security mechanisms, and this technique is cryptography [Stallings, 2003]. Among the security mechanisms are digital signature, access control, data integrity, authentication exchange, traffic padding, routing control, etc.

Table 1 shows the relationship between security services and security mechanisms, [Stallings, 2003]. In this table there is an explanation about what mechanisms are required to perform specific services. For example, a security service such as Data Integrity is achieved through the Enciphering and/or Digital Signature, while an Access Control Enforcement Policy is utilized to gain Access Control (security service). In Stallings' table (Table 1), there is no clear distinction between the services and mechanisms, however, any service may use one or more mechanisms to maintain or get the best possible results.

	Mechanism							
Service	Encipherment	Digital signature	Access control	Data integrity	Authentication exchange	Traffic padding	Routing control	Notarization
Peer entity Authentication	Y	Y			Y			
Data origin Authentication	Y	Y						
Access control			Y					
Confidentiality	Y						Y	
Traffic flow confidentiality	Y					Y	Y	
Data integrity	Y	Y		Y				
Non-repudiation		Y		Y				Y
Availability				Y	Y			

Table 1: Relationship between Security Services and Mechanisms

2.3 Security Service's Technologies

There are many different technologies, methods and techniques, which can be utilized to help us achieve our security service objectives such as data integrity, non-repudiation, availability, and data confidentiality, etc. The basis for most of the security services is Cryptography and its related technology. "Most agree that the basic theory of computer security is found in cryptography and network security protocols. There is also general agreement that the management of a computer security system involves security policies, disaster recovery, intrusion detection and monitoring of the entire enterprise's computers" [George, 2003].

2.3.1 Cryptography

In general and within the security countermeasures, cryptography has to achieve some security goals; such as Confidentiality, Data Integrity, Authentication, and Non-Repudiation. Cryptography targets among others these four different goals and adequately resolves the consequences of their applications. Two commonly used cryptography methods, which are symmetric and asymmetric (or public) key cryptography, will be discussed in the next subsection. Before this discussion, we will define some terminologies, which will be used here and thereafter [Stallings, 2003]:

- Plaintext: the original data that is entered into the cryptography system as input.
- Encryption algorithm: the algorithm that is going to substitute and transform the plaintext to an unreadable form of data.
- Encryption Key: the secret key, which represents a unique and independent value of the original message, and used as an input to the encryption algorithm. The output of the encryption algorithm will vary depending on the specific value of the key, different key's value will produce different out put.
- Ciphertext: This is the output of the encryption algorithm, and it represents a scrambled value of the original data. The ciphertext depends on the value of the

plaintext and on the value of the encryption key; in general, ciphertext is a random stream of data.

2.3.1.1 Symmetric-key Cryptography

Symmetric key cryptography is the type of cryptosystem where encryption between two or more parties is based on a single shared key, which is also used for decryption of the encrypted message between two or more communicating parties. This is different from public key cryptography, which uses two different keys; one for encryption and one for decryption of the data, see Figure 4 [Anish, 1997]

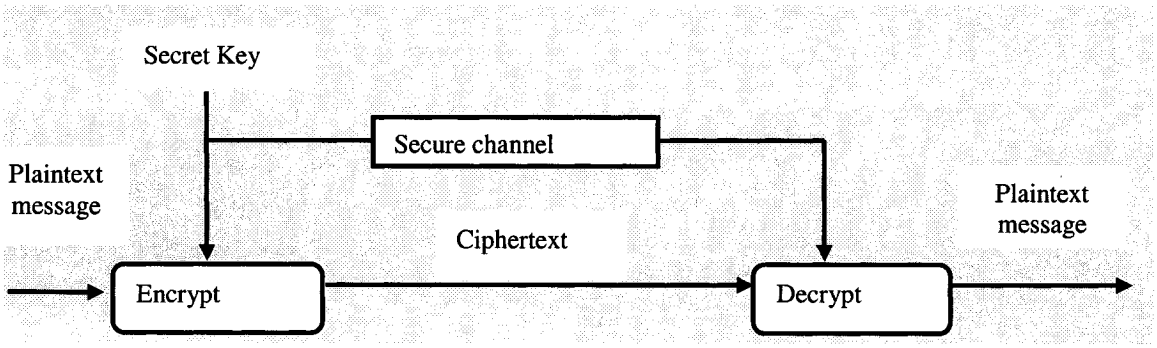


Figure 4: Symmetric Cryptography

There are two requirements for a secure use of symmetric encryption; first, a strong encryption algorithm is needed. At a minimum, the algorithm has to be strong as such that an attacker who knows the algorithm and has access to one or more ciphertexts is unable to decipher the ciphertext or figure out the key. Second, the sender and the receiver must have previously obtained copies of the secret key in a secure fashion and must keep the key secure. If the key is discovered and the algorithm is known, all communication using that key will become readable.

2.3.1.2 Public-key Cryptography

Public key algorithms use two keys to perform asymmetric encryption and decryption. That is, we use one key for encryption and the other for decryption. Public key cryptography succeeds only as long as a private key's owner keeps it under control, always available when needed but never disclosed to anyone else. Furthermore, public key encryption schemes are typically substantially slower than symmetric-key encryption algorithms "strong public key cipher are computationally very expensive, running usually 1000 time slower than comparable private key cipher" [Jerome, 2001]. For this reason, public key encryption is most commonly used in practice for the transport of keys whereas bulk data encryption are done by symmetric algorithms and other applications including data integrity and authentication, and for encrypting small data items such as credit card numbers and PINs. Public-key decryption may also provide authentication guarantees in entity authentication and authenticated key establishment protocols. One objective of an attacker who wishes to attack a public-key encryption scheme is to systematically recover plaintext from ciphertext intended for some other entity. If this is achieved, the encryption scheme is informally said to have been broken. For more on public key and crack down on cryptography see [Shai, 1999] [Garfinkel, 1996][Caloyannides, 2000]. Below we will discuss RSA public key cryptosystem, which is the commonly used public key cryptosystem.

The RSA public key Cryptosystem was named after its inventors, R. Rivest, A. Shamir, and L. Adleman. It may be used to provide both secrecy and digital signatures and its security is based on the intractability of the integer factorization. Various protocols and applications use RSA to authenticate people and computers, and to safely distribute keys used with private key encryption. An RSA key pair actually consists of three parts: the public part **e**, private part **d**, and the shared value **N**. We can construct the shared value **N** by choosing two very large prime numbers and multiplying them together. In practice, we often use a standard value for **e**, although the value of **e** can be chosen randomly as long as it meets certain mathematical restrictions. We compute **d** from **e** and the two large primes. The RSA public key consists of the shared value **N**

along with the public part **e**. Once we have erased the data used to produce the keys, only the private part **d** needs to be kept secret [Steve, 2001].

RSA encryption and decryption are surprisingly simple in design. If we are encrypting with the public key, then we take the plaintext message **P** and exponentiate it by the public part **e**, and take the modulus relative to **N**. In other words, we raise the number **P** to the **e**'th power, and then find its remainder relative to **N**. The modulus function simply computes the integer remainder. One common use of the RSA implementation of the public key cryptosystem is digital signatures, which is going to be discussed in the next subsection.

2.3.2 Digital Signatures

A digital signature of a message is like a real stamp or signature which depends on some hidden key known only to the signer. It also depends on the integrity of the content of the message being signed. Signatures do not need to be retrievable but they must be verifiable; that is if a dispute arises as to whether a party signed a document, a well known, trusted third party should be able to resolve the matter equitably, without requiring access to the signer's private key. Digital signatures have many different kinds of applications in the web security, including authentication, data integrity, and non-repudiation. One of the most important applications of digital signatures is the certification of public keys in very large networks. Certification is a mean for a trusted third party that can authenticate the identity of a user associated with a public key. Mainly, the digital signature is represented in a computer as a string of binary. A digital signature is computed using an algorithm and a set of parameters and authenticates the integrity of the signed data and the identity of the signatory.

The theoretical idea and utility of a digital signature was well known before any practical realization was available in the market. The first method introduced was the RSA signature scheme, which remains today one of the most practical and versatile techniques available. There are also many alternative digital signature techniques, some of which are offering significant improvements in term of functionality and

implementation. Figure 5 [Anish, 1997], explains the dynamics of using the public key to create a digital signature.

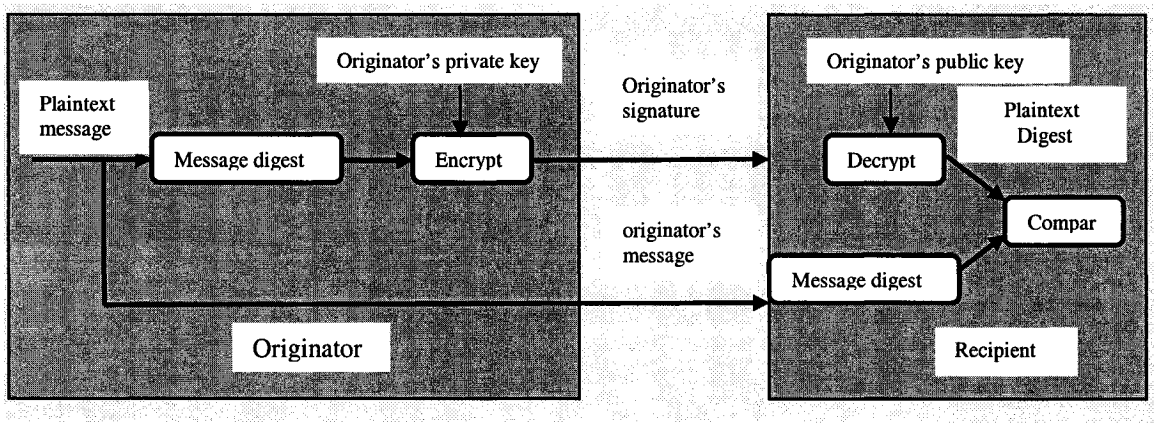


Figure 5: Using public-key systems to create digital signatures

2.3.3 Hash Functions

Cryptographic hash functions play a fundamental role in modern cryptography, and are used to provide the confidentiality and the integrity of the data and message authentication. The simplest form of hash functions is to take a message as input and produce a related output referred to as a hash code, a hash-result, a hash-value, or simply a hash [Alfred, 1996].

Hash functions are used for data integrity in conjunction with digital signature schemes. This kind of practice is used for different reasons. For example, a message is typically hashed first, and then the hash-value, as a representative of the message, is signed in place of the original message. Simply, this will make the message verifiable and not retrievable. A distinct class of hash functions that is being used today, called Message Authentication Codes (MACs), allows message authentication by symmetric encryption technique. MAC algorithms can be considered as hash functions that take two different inputs, a message and a secret key, and produce a fixed-size (n-bit) output, with the design intent that it be infeasible in practice to produce the same output without knowledge of the key. In this algorithm we can use a secret key and that is why the

message authentication codes can be used to provide data integrity and symmetric data origin authentication, as well as identification in symmetric-key schemes. For more details about MACs and the attacks against it, refer to [MAC].

2.4 Biometrics Security Services

Biometrics uses personal features instead of mathematical numbers and/or tokens to authenticate a person [Satyanarayanan, 1989]. Biometrics authentication has been used for centuries in one form or another. Computers use special devices to perform the biometrics reading and use a computational procedure to verify the user's identity. The procedure compares a biometrics reading against a corresponding reading stored in that person's user record. If the two readings match closely, then the system authenticates the person. Unlike passwords, biometrics data rarely matches perfectly. Instead, the authentication mechanism measures how close the latest reading matches the reading in the user records. Readings from the right person are supposed to match closely while reading from other people should not match.

2.4.1 Biometrics Fundamentals

Deploying biometrics effectively requires a solid understanding of the fundamentals of the technology. While leading biometrics technologies vary in complexity, capabilities, and performance, there are a number of elements shared by all of them. Template generation matching and error rates, and enrolment processes are among the many concepts central to biometrics that has a significant impact on system design, deployment costs, and individual privacy.

For instance, in the biometrics-based authentication, there is a wide range of sensors and measurements used for biometrics, all of which share the same fundamental design. All systems use a specific device that collects digitized measurements of a personal entity. An iris pattern system uses a specialized camera that focuses specifically on the iris and collects its image. Fingerprint systems typically use specially designed scanners to collect the fingerprint, and scanners intentionally do poorly at detecting patterns in other types of surfaces [Biometric Authentications].

A biometrics system performs feature extraction on the digitized data to identify the distinctive features associated with the particular biometrics. In a fingerprint system, for example, features consist of splits and junctions in the print's concentric ridges, combined with information about the features' relative locations. The extraction process yields a data item called the biometrics signature. To authenticate a particular user, the system looks up the user's record, retrieves the biometrics pattern associated with that user, and compares the derived signature against that pattern. The comparison process is generally designed to expect partial matches and to specify the degree to which the signature matches the pattern. If the two matches closely enough, the system authenticates the user to get access into the system, otherwise the access demand will be rejected, see Figure 6.

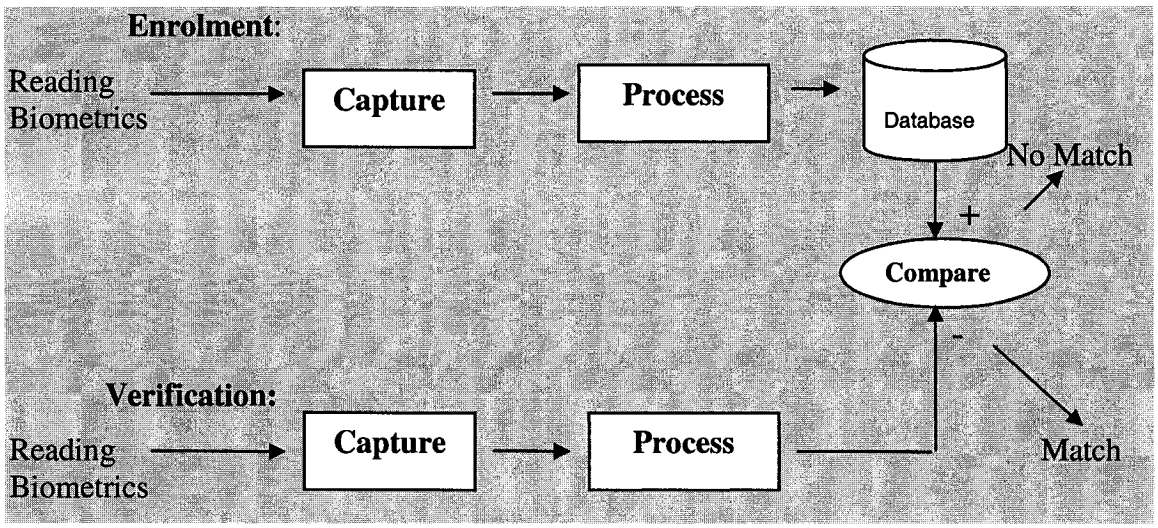


Figure 6: Biometrics Security Service Model

One of the most important problems with the biometrics security process is the sniffing attack. In this attack, the attacker tries to steal the biometrics information while it is in transmission and replay it later. We will discuss some of the countermeasures to defend against this sniffing attack in the next section.

We can use a public key with a biometrics system by embedding the public key in all biometrics readers. Whenever a biometrics signature is collected and sent in for authentication, we can encrypt it first with public key. This will prevent attackers from sniffing the biometrics; in fact, this approach would reduce the risk of someone attempting to intercept a biometrics reading collected by the system.

The use of public key algorithm can also authenticate the source of a biometrics reading. Private keys can also be embedded in biometrics readers. The reader's key would be used to digitally sign the biometrics so that the system has confidence that the reading originated from an acceptable reader. This gives the system a way to detect spoofed biometrics, if we assume that the readers themselves can reliably detect spoofing. Of course, this approach gives rise to the problems of managing the private keys that must be installed in all of biometrics readers. If the proprietor is willing to spend the administrative resources that are necessary to manage those private keys, it may make more sense to simply assign the keys directly to individuals instead of assigning them to biometrics readers. This is an architectural trade-off that individual sites and system managers must make.

While we can use the SSL's approach as public key cryptography to protect biometrics readings from interception, we cannot use it to prevent spoofing. For that, we need to associate a base secret with the transmitter of the biometrics reading, so we are sure the reading originates from a trustworthy source. Otherwise, an attacker with a copy of the public key and a convincing version of the victim's biometrics reading can masquerade as that victim. A list of possible security attacks is presented in chapter 4.

CHAPTER III

The Security Taxonomy Methodologies

In order to systematically develop our taxonomy, we are going to analyze, assess, and categorize some of the algorithms and methods that are being used for today's security taxonomy. This analysis and assessment will provide a comparative evaluation of the risks and benefits of each of these algorithms, and will show how those taxonomies differ from our taxonomy which will be presented in chapter 4.

3.1 Security Attacks Taxonomy

Studying and analyzing a wide range of security attacks taxonomies, is essential in assessing the overall system security. Security attacks taxonomy provides better understanding of the flaws and vulnerabilities of the security system. Moreover, security attacks taxonomy helps software designers and developers to derive constrictive methods in building security countermeasures. Following, we present some of the proposed security taxonomies, along with brief assessments of their performance.

3.1.1 Class of Threats Taxonomy

One of the important works in defining a good taxonomy has been accomplished by Stalling [Stalling, 95]. In this work, Stalling presented what he calls "A Process-Based Taxonomy."

Stallings presented his taxonomy as class of threats. He anticipated the objectives of an attacker by applying the process of the attack. For example, the process of an interruption will lead to the unavailability of the system or information, which comes as a result of a different rational of attacks. In his approach, Stallings concentrate on the security threats during the transmission of data over the Internet, which presents only a subset of the Internet security. Following are the classification categorizes for Stallings's taxonomy [Stallings, 95].

i. **Interruption:** An asset of the system is destroyed or becomes unavailable or unusable.
ii. **Interception:** An unauthorized party gains access to an asset.
iii. **Modification:** An unauthorized party not only gains access to, but tampers with an asset.
iv. **Fabrication:** An unauthorized party inserts counterfeit objects into the system.

This classification can be performed under a wider framework. For example, interception can be viewed as a passive attack, while interruption, modification and fabrication can be considered as an active one. This taxonomy has a very broad framework with an unspecific and very general classification which may consider enough for assessing the rational of the attacks, but the reality check shows that there is a huge gap of missing information that is required to serve the cause of the taxonomy as we will explain that in chapter 4.

3.1.2 Software Security Faults Taxonomy

A different taxonomy, by Aslam, Eugene, and Spafford [Aslam, 1996] was developed with the aim to be more suitable for data management and data organization from the security point of view. There is a common concept between this approach and the security flaws taxonomy by Landwehr presented later in this chapter. Their approach was focused on the software development faults and the consequences that originate from faulty software development. Hence, the benefits of their taxonomy are restricted to one area of the security flaws. However, their broad classification of the faults was categorized as code faults, and emergent faults. Obviously, this taxonomy discards some of the essential aspects of the security assessment. Buggy software can cause security flaws, but it is a software development issue and it has to be discussed and considered within those boundaries. More specifically, software development considerations have to fall outside the classification of the security threats, unless they are imposed on the system from the outside when attackers exploit the vulnerability embedded within the

software. Our taxonomy presented in Chapter 4 will consider these issues and differentiate between software bugs and malicious programs such as viruses, time bombs, etc. Following are the classification categories of the Software Faults Taxonomy [Aslam, 1996]:

- **Coding Faults**
  - Synchronization errors,
  - Condition validation errors,
- **Emergent Faults**
  - Configuration errors,
  - Environment faults,

3.1.3 Two-Dimension Taxonomy

One of the important works in classifying security attacks was done by Perry and Wallich [Perry, 1999]. Their work presented a security attacks taxonomy that has two-dimension specifications [Perry, 1999]. The first dimension was the potential perpetrator such as operators, programmers, data entry clerks, internal users, outside users, and intruders. The second dimension was the potential effect such as physical destruction, information destruction, data diddling, theft of services, browsing, and theft of information. As we can see in Table 2 [Perry, 1999], each and every cell represents a combination of the two dimensions in this matrix. They range from theft of data to entry of false data, unauthorized action, malicious software, etc.

The idea behind this matrix is to widen the scope of the classification and form a map that matches potential attackers to the potential damage, such a map by nature is not logical, because it is not possible to associate potential attacks to specific damage. In fact this is one of the limitations of this taxonomy, because creating such a map assumes a restriction of specific kind of damage to a specific kind of attacker and vice versa. For example, it is not logical to restrict the physical destruction only to the operator, or restrict the information destruction to the operator and programmers. In our taxonomy, we will give special attention to this point. We will introduce complete classification

categories of the attacks with an exhaustive map to the security services. Following is the two dimension matrix presented by Perry and Wallich [Perry, 1999].

	Operator	Programmer	Data Entry	Internal	Outside	Intrude
Physical Destruction	*Bombing Short circuits*					
Information Destruction	*Erasing Disks*	*Malicious software*			*Malicious software*	*Via modem*
Data Diddling		*Malicious software*	*False data entry*			
Theft of Services		*Theft as user*		*Unauthorized action*	*Via modem*	
Browsing	*Theft of media*			*Unauthorized access*	*Via modem*	
Theft of Information				*Unauthorized access*	*Via modem*	

Table 2: Two-Dimensional Security Attack Matrix

Generally, this matrix represents an improvement over the one dimension taxonomy approach because it is two-dimensional with respect to security attacks and damages, but still, the cells of this matrix cannot cover the whole area of security attacks as it is going to be driven in our taxonomy. More importantly, the link and the map between the two dimensions in this matrix are not properly formed.

3.1.4 Three-Dimension Matrix Taxonomy

Another approach which produced three-dimension matrix taxonomy was presented by Landwehr [Landwehr, 1994]. This work was focused on presenting taxonomy as a security flaw that can result in different forms, such as denial of service or unauthorized access to information. For more reading on protection against unauthorized access, see [Moldovyan, 2003]. The base of this matrix's three dimensions are; Genesis (how a security flaw finds its way into a program), Time of Introduction (in the life cycle of the

software or hardware), and Location (in software or hardware) [John, 1998]. This approach does not focus on the results or the effects of the security attacks as major classification characteristic but rather on the flaws of the attacks and the means they use as we will discuss in the following paragraphs.

3.1.4.1 Security flaw taxonomy: Flaws by Genesis

By analyzing Landwehr's taxonomy as presented in Table 3 [Landwehr, 1994], several points can be derived. First of all, we can conclude that a security flaw may be introduced into a program either intentionally or inadvertently. Secondly, to differentiate between accidental flaws and those intentionally inserted and to devise countermeasures to these flaws, the author is suggesting different methods and tests. For instance, increasing the effort of test and code review can reduce the accidental flaws and at the same time, trace the accidental flaws through different levels of the software development stages. So, reviewing the high level and the detailed design level with the security concern in mind will help to detect and reduce accidental flaws. On the other hand, if flaws are introduced intentionally in to the program, then tough testing measures, code and design review will not be that helpful. Therefore, the intentionally inserted flaws have to be treated as malicious software and all the necessary measure must be taken, such as some means of virus detection and a firewall. The overall advantage of differentiating between accidental flaws from intentionally inserted flaws is to formulate a knowledgeable decision regarding security countermeasures. The trickiest part in this can be the intentionally inserted features that have a dual use, where one user is secure and the other is not.

One of the major weaknesses of Landwehr's taxonomy is the ambiguity of the classification itself. For example, the attempt of classifying the virus as a Trojan horse is not universally accepted.

<table>
<tr><td rowspan="8">Genesis</td><td rowspan="5">Intentional</td><td rowspan="3">Malicious</td><td rowspan="2">Trojan Horse</td><td>Non-Replicating</td></tr>
<tr><td>Replicating (virus)</td></tr>
<tr><td colspan="2">Trapdoor</td></tr>
<tr><td colspan="2">Logic/Time Bomb</td></tr>
<tr><td rowspan="2">Non-Malicious</td><td rowspan="2">Covert Channel</td><td>Storage</td></tr>
<tr><td>Timing</td></tr>
<tr><td colspan="3">Other</td></tr>
<tr><td rowspan="5">Inadvertent</td><td colspan="3">Validation Error (Incomplete/Inconsistent)</td></tr>
<tr><td colspan="3">Domain Error (Including Object Re-use, Residuals, and Exposed Representation Errors)</td></tr>
<tr><td colspan="3">Serialization/aliasing</td></tr>
<tr><td colspan="3">Identification/Authentication Inadequate</td></tr>
<tr><td colspan="3">Boundary Condition Violation (Including Resource Exhaustion and Violable Constraint Errors)</td></tr>
<tr><td colspan="3">Other Exploitable Logic Error</td></tr>
</table>

Table 3: Security flaw taxonomy: Flaws by Genesis

3.1.4.2 Security flaw taxonomy: Flaws by Time of Introduction

Landwehr's taxonomy dealt with another dimension which takes a different approach to the categorization of the security flaws. It introduces timing classification. This means that it is concerned with the time that the security flaws are inserted into the program. In general, software development has many different concurrent or parallel levels of development such as requirements and specifications, design, implementation, test, integration, and operation. The rational behind this approach is to improve the software development process from the security point of view. To achieve the goal of

finding flaws in the software, a wide rang of software development models have to be studied to allocate all the points that are being considered in this taxonomy.

To be more specific, there are three phases in the system life cycle when security flaws may be introduced; the development phase, the maintenance phase, and the operation phase. The development phase includes all the systematic processes from the specification up to the deployment of the system. The maintenance phase includes all the activities that can provide a mean to adjust, modify and improve the performance of the system after the initial operation. Finally, the operation phase includes the adaptation and insertion of any kind of flaws during the operation time of the system. There is obviously an overlap between the maintenance and the operational phase, but still they are distinct enough to fulfill the requirements and serve the cause of the taxonomy, that is, to be specific and provide a method to countermeasure the flaws. In our taxonomy we deal with a software flaw as one category but we make a very clear distinction between software flaws, bugs, virus, and so on.

<table>
<tr><td rowspan="5">Time of Introduction</td><td rowspan="3">During Development</td><td>Requirement/ Specification/ Design</td></tr>
<tr><td>Source Code</td></tr>
<tr><td>Object Code</td></tr>
<tr><td colspan="2">During Maintenance</td></tr>
<tr><td colspan="2">During Operation</td></tr>
</table>

Table 4: Security Flaw Taxonomy: Flaw by Time of Introduction

3.1.4.3 Security flaw taxonomy: Flaws by Location

The third dimension of Landwehr's taxonomy tries to classify the flaws according to their location in the system. The idea is to be very specific about the performance of the flaws and how they affect the overall performance of the entire system. The flaws can be in the software or the hardware. Some times, flaws located in the software can cause flaws in the hardware. Flaws in the software may reside at different levels such as the operating system; application; and/or communication protocols. Hence, locating the flaws may help find a suitable countermeasure to prevent or limit the security attacks that could exploit the vulnerability of any flaw. This taxonomy adopts this approach to specifically address the problems that originate in software bugs and define their impact on the security of the system.

<table>
<tr><td rowspan="10">Location</td><td rowspan="9">Software</td><td rowspan="7">Operating Systems</td><td>System Initializations</td></tr>
<tr><td>Memory Management</td></tr>
<tr><td>Process Management / Scheduling</td></tr>
<tr><td>Device Management (including I/O, networking)</td></tr>
<tr><td>File Management</td></tr>
<tr><td>Identification/Authentication</td></tr>
<tr><td>Other / Unknown</td></tr>
<tr><td rowspan="2">Support</td><td>Privileged Utilities</td></tr>
<tr><td>Unprivileged Utilities</td></tr>
<tr><td colspan="2">Application</td></tr>
<tr><td colspan="3">Hardware</td></tr>
</table>

Table 5: Security Flaw Taxonomy: Flaw by Location

From table 5, one can realize that this approach considers the operating system as a major target in assessing the security flaws. However, this kind of assessment does not

mean that the software flaws are exclusive to the operating system. In fact, the application software has its own serious flaws that still need to be structured and categorized. Hence, the author of the taxonomy leaves the classification part as future work, thus encouraging other researchers to take initiative steps towards producing a classification category for the security flaws in the application programs. Allocating the security flaws in the system would eventually help in building a stronger system in the future. This concern will be thoroughly satisfied in our taxonomy.

3.1.5 Howard's Taxonomy

Table 6 presents the taxonomy that has been produced by J. Howard [John, 1998]. This work divides the attack into six steps that the attacker should follow in order to be successful. The attacker needs to follow one or more of the paths in this taxonomy. The idea behind the computer security services is to provide the means and technologies to prevent any attacker from reaching his/her goal. In order to assess and hence prevent the attack, this taxonomy defines six categories where the attack can be prevented.

In the first category, attackers are classified based on their performance and their objectives. Different attackers could be motivated differently, such as hackers, spies, professional criminals, vandals, etc. Hence, security professionals have to find ways to stop the attack by classifying, allocating and identifying the attackers. Unique measurements can be taken against different attackers based on who they are and their motives. Classifying the attackers and their motives may not be that helpful in devising the most suitable prevention strategies that the security experts may adopt. No doubt, attackers must be stopped; their tools and techniques must be detected, and counter measured, regardless of whom they are and what their motive is.

Howard's taxonomy [John, 1998] suggests taking security measures like firewalls and antivirus programs. Observing the system and maintaining a regular check on the saved and incoming data can detect the presence of Trojan horses, or any other unauthorized files. System administrators and security personnel are responsible for watching the overall performance of the system and detecting any cracking attempt. This helps them in analyzing the techniques and technologies the attackers use so they might provide

preventive countermeasures. Tools used to attack systems are becoming more aggressive, accurately targeting, and more sophisticated to a point beyond the scope of Howard's taxonomy.

The broad classification of the access to the systems has two categories: unauthorized access, and unauthorized use. System administrators have to set the privileges and restrictions for all legitimate users of the system. Authorized access to the system is not an attack, but achieving unauthorized objectives through authorized access can be considered one. To prevent authorized security abuse of the system, a system administrator has to establish control access enforcement policies and procedures that will restrict and reduce the access to the most sensitive data and information to a minimum, and will maximize the privileges to trustworthy users.

The risk is resurrected when a legitimate user may misuse or even abuse the access by different means. In such a case, the system administrator has to be vigilant and implement mechanisms that can work as watch dogs and monitor the performance of the users to detect any current abuse and prevent future ones. On the other hand, the unauthorized access to systems has to be prevented by providing strong authentication mechanisms. Strong authentication mechanisms include many different technologies, techniques and procedures as discussed in Chapter 2. In general, attackers will exploit the implementation, design, and configuration vulnerabilities to gain unauthorized access or unauthorized use of the system. Howard's taxonomy classifies the access into three levels as can be seen in Figure 7 [Howard, 1996]. The access element of this taxonomy took advantage of the security flaws classification of Landwehr's and Alam's.

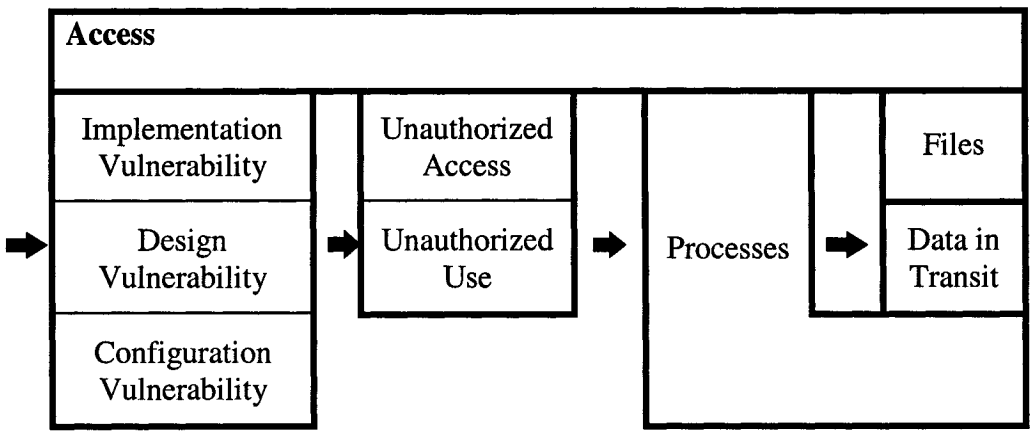


Figure 7: Access for Attack

This taxonomy, in general, differentiates between the result of an attack and the objective of the attacker, which may overlap in some cases, and that is not preferable in taxonomy. In this classification, the results were categorized as follows:

- Corruption of Information.
- Disclosure of Information.
- Theft of Service.
- Denial of Service.

According to Howard's Taxonomy, result is the third step in the stream of any attack. This means that the attacker is already gaining access to the system, whether it is authorized or not. The attacker is then in an advantageous position because he/she is getting through one of the most secured procedures in Internet security system "the authentication". For example, the attacker may perform an unauthorized modification or deletion of the data saved in either the server or the client. This is called corruption of information. The attacker might also be interested in only monitoring and unfolding confidential information, which this taxonomy calls disclosure of information [John, 1996]. Other results may be unauthorized use of the service, blocking the whole service, or/and preventing legitimate users from practicing their normal access to the service. We have found that this classification category is subset of the process-base taxonomy introduced by Stallings, and discussed in page 23.

The classification of the result in this taxonomy may fall short in covering all the possible results from any specific attack. For example, attacks by someone trying to send false information while masquerading as someone else are not covered. However in our taxonomy, such uses and results are going to be discussed within the security services. For that reason our taxonomy will also map each and every class of the security attacks to its security services.

In addition to that, Howard's taxonomy summarized the vulnerability in three components where the attacker can gain some advantages: design, implementation, and configuration. Classifying the vulnerability in this way is not completely practical. For example, if we take a hypothetical assumption and say that the design, implementation and configuration are 100% perfect, does that mean that there is no attack that can impose itself successfully on the system? It is difficult to agree on that, daily active attacks, using so many techniques that are not concerned with the design, implementation and/or configuration, such as the attacks that use social engineering as a tool. (For more on social engineering see [Mitnick, 2002]). In the next two steps there is no mention of the non-repudiation, when the attacker, for example, processes a financial banking order on behalf of someone else. Furthermore, the approach itself is not really convincing, because of many different reason. For example, part of the access been classified as an unauthorized access, and unauthorized use, and there is no mention whatsoever to the authorized user that may gain unauthorized results, moreover, and the utilization of such taxonomy approaches to zero in such a fast moving technology.

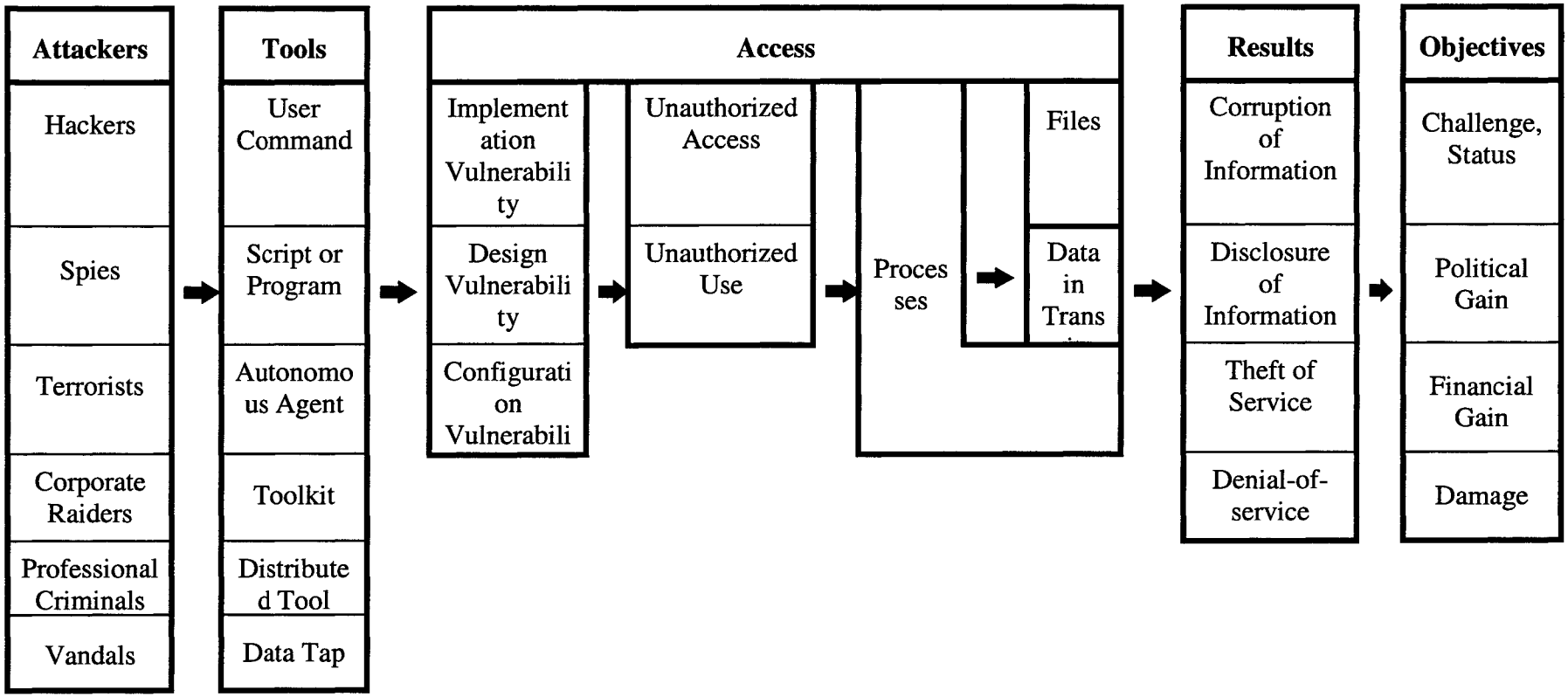


Table 6: Howard's Taxonomy

3.1.6 Neumann's List Taxonomy

The Empirical List Taxonomy [Neumann, 1998] developed by Neumann and Parker tries to assess and classify accounts of actual attacks sent to Neumann at Stanford Research Institute (SRI) International laboratory as part of its Risks Forum ("Risks to the Public in Computers and Related Systems"). There are eight categories being used to classify their data. The main advantage is that this classification has enough categories to cover a wide range of possible security attacks. But still, there is an overlap between the categories, which means the classification is not exclusive. For example, masquerading may use a technique to defeat authentication or authorization service which may cause an overlap between two different categories.

Following is the Neumann and Parker list [Neumann, 1998].

- **External Information Theft**: Monitoring information at the other user terminal and/or personal note.
- **External Abuse of Resources**: Physical damage of resources.
- **Masquerading**: Replaying a data transmitted previously, and/or pretending to be someone else.
- **Pest Programs**: Installing a program that has the ability to inflict certain levels of damage.
- **Bypassing Authentication or Authority**: The attempt to crack into the system without having the right authority.
- **Authority Abuse**: Attack coming from insiders.
- **Abuse through Inaction**: Careless administration,
- **Indirect Abuse**: Transmitting a program that has the ability to cause a certain level of damage.

Taking into consideration the simplest and yet, most common definition of a security attack [John, 1998], which is an unauthorized access and/or unauthorized use of the system and its components; the empirical list taxonomy has some drawbacks. For example, Abuse through Inaction, in most general cases can not be considered as an attack because careless administration may cause a problem not only in security but also

in most of the system's aspects and utilization. Furthermore, bad administration is not an attempt to gain unauthorized use or unauthorized access into the system.

Another misunderstanding in this taxonomy is the External Abuse of Resources, which comes in the form of a physical damage of resources. It is improper to consider a physical abuse or damage as a security threat unless it is very specific and targeted. For example, an earthquake may cause a physical damage but it is really incorrect to consider it as a security threat, but if someone tries to break the hard drive do make the system of its information unavailable, this may pose a threat, therefore a clear distinction is required. The security experts can distinguish between physical properties of the system and the hardware that compose the communication system, such as networking processes, data bases, data transition, and communication devices. Such a distinction can provide a differentiation between a normal and a natural physical damage and a security threat. The difference between our taxonomy and this one is that we carefully articulate what the security threats are, whereas the Empirical list does not clearly distinguish between a security threat and any other type of malfunctions. As we explained previously, the empirical list has an overlap between its classes.

3.2 Security Service Taxonomy

To the best of our knowledge, there has not been a comprehensive work to classify the security services in conjunction with the security attacks. One of the reasons for that can be due to the fact that the whole topic of Internet security is relatively new and requires plenty of efforts and resources. However, there has been more research done on the area of security attacks taxonomy. One of the recognizable attempts to produce a security service taxonomy, that we are aware of, is "Toward a Taxonomy and Costing Method for Security Services" by C. Irvine, and T. Levin [Irvine, 1999]. This work can be considered as a preliminary approach toward a taxonomy that does not have that much to say about any specific requirement for the Internet security services and the attacks that can violate the services in the taxonomy. The main objective of the Irvine and Levin work is to present taxonomy of security services that can be used as framework to define

the quality of security services required by the Resource Management Systems (RMS), see Table 7 [Irvine, 1999].

In their work, the security services have been divided into three different areas that can reflect the general topographical structure of the networks where the security services are operating effectively. Accordingly, the security services were classified as: end system (such as client or serve), intermediate node (such as Hub, networks backbone), and the network connections (such as network cards, connectors, wires). The security mechanisms operating in the end systems can protect the resources and stored information. For the Internet security services, we are more interested in securing the information while in transmission. As shown in Table 6 [Irvine, 1999], each security mechanism is associated with a particular service area with specific notation used to reference the area: "**IN**" for Intermediate Node, "**W**" for wire and "**ES**" for End System. The Total Subnet (TS) service area identifies mechanisms that cannot be assigned exclusively to either IN, W, or ES.

According to the authors themselves, their work does not take into consideration any security attack that may exploit the vulnerability of the security services in order for the researchers to assess the efficiency of the security service and the completeness of the taxonomy itself. "The list of mechanisms, in that taxonomy, is not intended to be exhaustive; in fact, it provides a framework for illustrating the taxonomy" [Irvine, 1999]. In our taxonomy we will specify the functionality and performance of the security services with respect to the attacks and illustrate how the security countermeasures may improve the security services in any particular area.

SECURITY SERVICE	SERVICE AREA	EXAMPLE SECURITY MECHANISMS
Data Confidentiality	IN	OS access controls, Cryptographic credentials
	W	40-bit DES, 128-bit Blowfish
	ES	OS access controls, Cryptographic credentials
Traffic Flow Confidentiality	IN	Active network nodes monitor traffic and inject dummy packets in response to certain triggering conditions.
	W	communications uses a Virtual Private Network with encapsulated packets
	ES	Traffic padding up to a defined maximum is provided. Beyond that maximum, traffic flow confidentiality cannot be guaranteed
Data Integrity	IN	OS access controls, Cryptographic credentials
	W	Cryptographic chaining, integrity sequence numbers, and digital signatures
	ES	OS access controls, Cryptographic credentials
Authenticity	IN	Active network supports internodes authentication based on digital signatures.
	W	Data origin authentication, i.e. IP address, digital signatures
	ES	OS identification and authentication mechanism; use of Digital Signature Standard; use of trusted certificate authority
Non-Repudiation	IN	Active network nodes report transactions to secure logging facility.
	W	digital notary and non-repudiation services
Guarantee of Service, Availability	IN	Active network nodes reserve bandwidth for network administrative traffic. Priority-based scheduling for application traffic.
	W	Bandwidth reservation protocol.
	ES	Time-slicing scheduler, FIFO scheduler with pre-emptive interrupts.
Audit and Intrusion Detection	IN	auditing of network control functions
	TS	rule-based and profile-based network intrusion detection, intrusion correlation engine to identify intrusions across a group of subnets
Boundary Control	TS	firewall, proxy server, guard

Table 7: Preliminary Security Service Taxonomy

CHAPTER IV
A New Internet Security Taxonomy

Security in Internet systems is essential and crucial so as to ensure secure, reliable and available operation and to protect the stored information and/or the information while it is in transmission. Security vulnerability in any components of the Internet system can be exploited to breach the security of the system. Security vulnerability has to be traced down, identified, classified, and countered to ensure reliability and safeguard against any attack such as unauthorized modification or usage of data or information.

In this chapter, the focus will be on developing taxonomy for the types of security services and attacks. Categorizing actual and possible Internet security attacks will be introduced. In the next chapter, the attacks within each category and existing solutions that deal with them will be identified.

4.1 Classifying the Base Technology of the Security Mechanisms.

In general, and as discussed in chapter 2, security services can be classified based on the technologies. Later in this chapter, security services will be classified based on the objectives that the security service would provide. We will therefore produce a list of categories for most of the identified security attacks.

4.1.1 Cryptographic and Biometrics based security services

According to the major technologies which initiate the process of the security services, the security services will be classified into two categories:

1. **Cryptographic-Based security services**
2. **Biometrics-Based security services.**

To understand the characteristics of both cryptographic-based and biometrics security services we compared some of their benefits and weaknesses as presented in Table 8.

Internet Security Service	Benefits	Weaknesses	Examples
Cryptographic Based	• Cheap to implement • Portable • Some time is hardest to abuse	• Sniffing attacks • Passwords are either easy to guess or hard to remember • Cost of handing forgotten passwords	• Passwords • PIN • Safe combination
Biometrics Based	• Hard to implement • Easier to authenticate • Portable	• Replay threats • Expensive • Privacy risks Characteristic can't be changed false. Rejection of legitimate users • Characteristic can be injured	• Finger print • Eye scan • Voice recognition

Table 8: Biometrics and Cryptographic based security

Biometrics is much harder and costly to implement, but at the same time provides greater security and reliability in the authentication process and Internet access control. However, our research has shown that the biometrics-based security services could not become more secure than the cryptographic based security service if it is utilized alone, because biometrics has to be digitized and transmitted and therefore it may be vulnerable to the attacks. However, a system can become more secure, especially with respect to the authentication service, if biometric is combined with some sort of cryptographic algorithms. For example, operating a biometrics authentication system such as the one that uses fingerprint access to the Internet services can be strengthening. Embedding a cryptography method within a biometrics technique, and encrypting the biometrics information before transmitting or storing them can also lessen the possibility of the theft

of biometrics information. As a result, the system becomes less vulnerable by combining the biometrics with cryptography. Cryptographically-based and biometric-based security services are both going to be addressed in our taxonomy as well as the security attacks against them.

4.2 Internet Security Taxonomy

In this research we present the Internet security taxonomy, which consists of a two-dimensional matrix. One dimension represents basic security service characteristics that any reliable Internet security service should provide.

4.2.1 Articulating the Security Services

According to most of the literatures, especially the National Institution for Standards and Technology (NIST) [NIST, 2001], we have articulated the security services according to: Confidentiality, Data Integrity, Authentication, Authorization and Internet Access Control, Non-repudiation, and Availability.

In the following each of theses characteristics will be discussed in more details.

1. **Confidentiality** is the protection of transmitted data from being disclosed to any unauthorized party. With respect to the content of data transmission, several levels of protection can be identified. The broadest service protects all transmitted data between two users over a period of time. For example, if a TCP connection were set up between two systems, this broad protection would prevent the release of any user data transmitted over the TCP connection [Anonymous, 2001]. The other aspect of confidentiality is the protection of traffic flow from analysis. This requires that an attacker will not be able to observe the source and destination, frequency, length, or other characteristics of the traffic on a communications facility.

2. **Data integrity** can apply to a stream of messages, a single message, or selected fields within a message. The most useful and straightforward approach is the total stream protection, because it is going to protect the whole transmitted data. A data integrity service assures that messages are received as sent, with no duplication, insertion, modification, reordering, or replays. The destruction of data is also covered under this

service. Thus, the data integrity service addresses both message stream modification and denial of service. Providing data integrity services is an important task to ensure the privacy, security, and reliability of critical information. The data integrity can come under attacks and can be violated using different means and tools [Tulloch, 2003]:

- Corruption of data resulting from software bugs or the actions of malicious users
- Viruses infecting computer systems and Trojans masquerading as genuine applications
- Hardware failures caused by age, accident, or a natural disaster (which is not an attack).
- Human error in entering, storing, or transmitting data over a network (it may be considered as an attack if it is an intentional error).

3. **Authentication** is a service that can uniquely identify end users. The authentication process is applied to both, entities and information. Two parties participating in communication activities must identify each other. Items, entities, and or information delivered over a communication link should be also authenticated as to their origin. For these reasons, this aspect of cryptography is usually subdivided into two major classes: entity authentication and data origin authentication. Data origin authentication implicitly provides data integrity (if a message is modified, the source has changed) [Niels, 1998] while entity authentication is a process to authenticate a person or an organization. In general, information has to be genuine rather than modified or fabricated. So the information is authentic if it is identical to that created, placed, stored, or transferred as it is in the original copy.

4. **Authorization and Internet Access Control** is the mechanism of providing permission or privilege to any entity, such as person or organization, to have access to information or to use the information in a certain way. In a distributed system, such as the Internet, where many users share the same resources, system administrators must organize, synchronize and control the performance of the users. A System administrator has to allocate privileges and sometimes time slots for the users to access the system and its components. In the Internet systems, there is a real need to limit and observe the access to the host or Internet server and its applications through the communication

protocols. To process such control perfectly, each party trying to reach an access a specific system must be identified and authenticated, so that access privileges and rights can be tailored.

5. **Non-repudiation** is the process that aims to eliminate an entity from denying previous commitments or actions. Non-repudiation has two parts, origin and destination. At the origin side, non-repudiation proves that the specified party sent the message. At the destination side, non-repudiation proves that the specified party received the message. So, in general, non-repudiation provides protection for all the parties involved in the communication activity. When a dispute happens between different parties due to an entity denying that certain actions were taken, a means to resolve the situation is necessary. For example, one party may issue an authorization for purchasing property or company shares to some other party, and later denies such an authorization was requested. Such a case typically requires the need for the involvement of a trusted third party to resolve the dispute [Niels, 1998].

6. **Availability** is a process that provides the authorized users with the ability of gaining access to the appropriate system or information without any restriction or delay. An authorized user is not only the specified individual but also include all the parties involved in legitimate communication activities, such as organizations, systems. Obviously, availability dose not mean that the information or system has to be available to anyone anytime, but the processes of identification and authorization are highly required. The availability of information is guaranteed to the authorized users whenever they request them and in the required format. The core of the availability services lies in the ability of the system to establish effective and continuous operations. The major approach to provide effective and continuous operation is to protect the system against unauthorized users. Defining authorization and access control policies and finding a means to enforce such a policy could be the first step to defeating unauthorized users. One more important step is providing a reliable and available system that sets a highly sophisticated mechanism that can distinguish between authorized and unauthorized users. There are other means to maintain an effective and continuous operation such as

intrusion, detection, maintaining data recovery strategy, detecting unauthorized use of information.

4.2.2 New Classification Category of the Internet Security Attacks

The second dimension in our matrix taxonomy is based on the new classification categories of the Internet security attacks that we have developed. This classification is presented as a list of categories, which represent the actual and potential security attacks that may target the system. Our list of categories is formulated to present a framework where all the possible security attacks can be processed. The objectives and the affected areas of the Internet security attacks will be identified, and countermeasures will be introduced in the next chapter. A list of the proposed new classification categories of the Internet security attacks is given below followed by a deep explanation of each of theses categories:

- **Manual Penetrating of the System and/or Individuals Privacy;** (Includes all methods of password cracking, social engineering, masquerading etc);
- **Data Interception, Interruption, and Replaying**; (Includes interception of information and/or a sequence of communication process, tampering, modifying, message deleting of data while it is in transmission);
- **Biometrics and Physical Token** (Includes all attacks using token, biometrics , biometrics methods and processes etc);
- **Defeating Mechanisms and Policy**; (Includes all the challenges presented to the authentication, authorization, and access control mechanisms and policies);
- **Malicious Code** (Includes malicious software, viruses, malfeasant code, bugs, coding problem etc.);
- **Distributed Communication Systems** (Includes all different types of DoS, DDoS, and other attacks using a possible communication protocol as means such as the TCP/IP);

4.2.2.1 Description of the Newly Developed Categories.

1. **Manual Penetrating of the System and/or Individual Privacy:** This security attack category includes all methods of password cracking, stealing and using someone else's identity. Social engineering is also included in this category. Social engineering is the process of using social skills to deceive people to expose their secrets, such as passwords, or any other valuable information. Masquerading, which is articulated when someone is pretending to be another person would be included under this category. The masquerading technique would enable the attacker to bypass the authorization and control access restrictions to gain unauthorized access or more privileges.

2. **Data Interception, Interception and Replaying:** Interception, in general, is an unauthorized access to a system of information. In that order it may be viewed as a passive attack. An attacker could passively intercept information saved in the system or transmitted over the net. This will enable the attacker to analyze the communication traffic or expose sensitive data. An attacker may interrupt the information and perform some changes or deletion on it, which is considered as an active attack. At the same time, an attacker may retransmit the information to gain some advantages such as financial reward. An attacker may interrupt a sequence of communication processes or encrypted information to gain unauthorized access to the system. In this category, the attacker may modify or delete the data, whether it is saved or while it is in transmission. Performing delay on the transmitted data is one form of this attack. This attack may include any interference with the communication or messages over the net, especially when the message is being altered or modified.

3. **Biometrics and Physical Token:** Biometrics use biological features instead of facts for security process. Using the biometrics techniques is getting more attention, so are the attacks against biometrics. There are many different biometric mechanisms, and as we explained earlier, biometrics are associated with physical traits, such as finger prints, face recognition, voice recognition, retina scan, etc. This category covers the attacks against biometrics, such as copy biometrics signature, forged biometrics,

biometrics disclose, etc. This category also includes attacks against physical tokens, such as stolen smart cards.

4. **Defeating Mechanisms and policy:** Authentication schemes are one of most important parts in ensuring a secure system. There are many different schemes and mechanisms used in today's authentication process, but still, the most common and widely used ones are password-based. The attack against authentication processes is one of the most widely practiced methods of attacks. When defining authentication as a process of identification of the user, organization or process, the attack against them involves circumventing the identification process, one way or the other. Authorization is assigning privileges to any individual and entity. So, the attack against authorization may target the policies or may target the system administrator as well. Targeting the policies that have been set by the system administrator will result in gaining more privileges for unauthorized entitle, which can endanger the privacy or cause misuse of information.

5. **Malicious Code:** In this category, various kinds of attacks have been covered. For example, a virus, that is program developed to spread and infect the system and its operating system utilities and then make copies of itself. All other type of malicious software, bugs, worms, are covered under this category. Coding errors and flaws such as those introduced during different levels of the software development process are also covered.

6. **Distributed Communication Systems:** This category includes denial of service and distributed denial of service, and any other attacks using the Internet communication systems such as IP spoofing attack as means to achieve the attack objectives. The major objective of the denial of service attacks is to prevent the attacked victims from gaining access to any resources in the specific target, including resources or tools that may enable the victims to countermeasure such an attack. In general, a denial of service attack is classified in a wider framework as a prevention attempt to stop an authorized user of accessing a service that is restricted. A denial of service attack can take many different

forms and use different means. For example; flooding the communication system with thousands of messages to disrupt and prevent the legitimate traffic. Another example is to interrupt the connections between two systems or disrupt any specific system or individuals from accessing the services. While in distributed denial-of-service attacks (DDoS), many compromised systems have been synchronized to perform attacks on specific target. Therefore, the DDoS presents a real threat to the Internet, which is going to be covered under this category as well.

4.3 Developing a Comprehensive Internet Security Taxonomy

The first step toward developing our taxonomy is to build new classification categories for Internet security attacks, as presented in the last subsection. Secondly, after articulating the Internet security services, new and complete matrix taxonomy of security services will be presented. Analyzing security attacks and assessing the effected area of the attacks and linking them to the security services being introduced as possible countermeasure, is one of the benefits of our proposed taxonomy. This taxonomy provides a comprehensive view of the threats that may threaten the Internet's security, reliability and availability.

Our newly introduced classification of the security attacks is quite useful because it provides a very wide umbrella in which a large number of actual and potential attacks can be covered. We can include as many attacks under this umbrella as needed. In addition, our taxonomy satisfies the most important characteristics required by any classification category. Examples of those characteristics are being; mutually exclusive, exhaustive, unambiguous, useful and repeatable. Besides producing this taxonomy, the identification of the security countermeasure associated with each attack can be seen as another contribution of this research work. Our taxonomy specifies the nature of the security attack, the security service being violated by that specific attack and finally the security countermeasure to be used against such an attack.

In the Internet security services, we need to insure that the Internet security requirements are taken into consideration. Hence, in order for us to emphasize our exhaustive approach, we have expanded the security services that we have talked about, like: data integrity, data confidentiality, authentication, and non-repudiation into broader services. Therefore, our taxonomy includes the other security services, which could be needed specifically for the Internet security, such as the Internet Access control, Authorization and Availability. In that regard, we have a complete set of the Internet security services under which the performance of the Internet security system against different attacks can be analyzed.

4.3.1 New Two Dimension Matrix of the Internet Security Taxonomy

Table 9 shows our suggested new Internet security services taxonomy which is built as a two dimension matrix. One dimension represents the security services, and the other our new classification categories of the security attacks. For explanation about what the abbreviation in cells of the matrix represents, consulate the abbreviation table, in the end of the thesis.

Security Service / Class of Security Attack	**Confidentiality**	**Data Integrity**	**Authentication**	**Authorization & Internet Access Control**	**Non-Repudiation**	**Availability**
Manual Penetrating the System or the Individuals Privacy	C-MPSIP	DI-MPSIP	AUT-MPSIP	AIAC-MPSIP	NR-MPSIP	AV-MPSIP
Data Interception, Interruption and Replaying	C-DIIR	DI-DIIR	AUT-DIIR	AIAC-DIIR	NR-DIIR	AV-DIIR
Biometrics and Physical Token	C-BPT	DI-BPT	AUT-BPT	AIAC-BPT	NR-BPT	AV-BPT
Defeating Authentication Mechanisms and Policies	C-DAMP	DI-DAMP	AUT-DAMP	AIAC-DAMP	NR-DAMP	AV-DAMP
Malicious Program	C-MP	DI-MP	AUT-MP	AIAC-MP	NR-MP	AV-MP
Distributed and Communication Protocols	C-DCP	DI-DCP	AUT-DCP	AIAC-DCP	NR-DCP	AV-DCP

Table 9 Matrix of the Internet Security Taxonomy

The explanation of the cells of our taxonomy and their performance review with respect to the nature of the security attacks and countermeasures to those attacks is given below. Following are the tables presenting the cells of our taxonomy. Although the security attacks may become irrelevant because of the evolvement of the countermeasure techniques or improvement in the Internet technology, the classification categories and the class of each cells in the matrix of the proposed taxonomy might remain useful.

Table 10 shows the security service confidentiality vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM).

Mapped Entity Security Service/ Class of Security Attack	**Examples of SA/SCM**
C-MPSIP	Social Engineering/ Privacy and Unpredictable Password
C-DIIR	Eavesdrops/ Repeated Challenge Response
C-BPT	Biometrics Interception/ Biometrics Data Encryption
C-DAMP	Extract PIN/ PIN Incorporated into Base Secret
C-MP	Back Door/ Eliminate Back Door
C-DCP	Direct Communication Attack/ Observing and Restrict Connection to the System

Table 10: Confidentiality/Nature of Attack vs SA/SCM

Table 11 shows the security service data integrity vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM)

Mapped Entity Security Service/ Class of Security Attack	Examples of SA/SCM
DI-MPSIP	Passwords File Theft/ Hashed Passwords File
DI-DIIR	Modifying Intercepted Message/ Encryption System
DI-BPT	Forged Biometrics/ Encoded Passwords
DI-DAMP	Attacking Encryption Procedures/ Suitable Encryption Procedures
DI-MP	Trojan Horse/ Firewall
DI-DCP	Source Address Forgery/ TCP Synchronization

Table 11: Data Integrity/Nature of Attack vs SA/SCM

Table 12 shows the security service authentication vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM)

Mapped Entity Security Service/ Class of Security Attack	Examples of SA/SCM
AUT-MPSIP	On-line Password Guessing/ Audit Bad Passwords
AUT-DIIR	Password Sniffing/ Encrypted Password
AUT-BPT	Replicating the Biometrics Signature/ Authenticate Biometrics Signature/
AUT-DAMP	Public Key Forgeries/ Public Key Certificate
AUT-MP	Trojan Login/ Change Passwords
AUT-DCP	IP Address Theft/ GPS Location Authentication

Table 12: Authentication/Nature of Attack vs SA/SCM

Table 13 shows the security service authorization and Internet access control vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM)

Mapped Entity Security Service/ Class of Security Attack	Examples of SA/SCM
AIAC-MPSIP	Off-line Password Search/ Forced Lengthy Trails
AIAC-DIIR	Reply Hashed Password/ One-time Password
AIAC-BPT	Defeating matching mechanisms / Minimizing matching score.
AIAC-DAMP	Forge Authorization Privilege/ Encrypted Access Connection
AIAC-MP	Mutual Trust/ Firewalls and Enforce Access Control
AIAC-DCP	IP Spoofing/ Unpredictable TCP Sequencing

Table 13: Authorization and Internet Access Control/Nature Of Attack vs SA/SCM

Table 14 shows the security service non-repudiation vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM)

Mapped Entity Security Service/ Class of Security Attack	Examples of SA/SCM
NR-MPSIP	Account Theft/ Enrolment in Person
NR-DIIR	Sniffing a Private Key/ Public Key on Smart Card
NR-BPT	Biometrics Sensor Disorientation/ Check and Maintain Biometrics Sensor
NR-DAMP	Convert Reject into Accept/ Keyed Hash Incorporating
NR-MP	Buffer Overrun/ Server Encapsulation
NR-DCP	IP Hijacking/ Integrity of the Host OS

Table 14: Non-repudiation/Nature of Attack vs SA/SCM

Table 15 shows the security service Availability vs the nature of the attack with examples of security attack (SA) and security countermeasure (SCM)

Mapped Entity Security Service/ Class of Security Attack	Examples of SA/SCM
AV-MPSIP	OS Substitution/ BIOS Password
AV-DIIR	Unauthorized Delete of Data/ Limited Access
AV-BPT	Steal the Biometrics Token/ Backup Emergency Processes
AV-DAMP	Synchronization Flood/ Connection Management
AV-MP	Time Bomb/ Firewall and or Anti-virus
AV-DCP	Distributed Denial of Service/ Observing the System Performance

Table 15: Availability/Nature of Attack vs SA/SCM

4.4 Assessment of Our Internet Security Taxonomy

Our two-dimensional matrix (table 9) of the Internet security taxonomy presents a comprehensive approach and offers a better performance than any other taxonomy such as the ones that were introduced in Chapter 3. Taxonomy based on objectives, processes, tools, or any other restricted views will not be complete without a complete assessment of the phenomenon of the security attacks and services. The mapping tables (tables 10 - 15) show that the most commonly known attacks are covered under the classes of our proposed taxonomy. Moreover, and based on the knowledge acquired through the study of related work, this is the first time that biometrics and their related tokens are clearly defined as a category in the classification of the security attacks. Attacks against biometrics have also been identified and countermeasures have been suggested for them.

An important advantage of the proposed taxonomy is the mapping between the security services to the security attacks that are associated with them. For example, an attacker might use a victim's password or trick the authentication mechanism to break

into a system. These kinds of attacks may penetrate a person's privacy by stealing a password, or penetrating a system by running a password cracker. At the same time, such an attack is a violation of the authentication process. So, it is classified within this taxonomy under the class of **AUT-PSI**. Our taxonomy offers a countermeasure for each such attack, thus improving the use and the performance of such taxonomy.

To illustrate this further, consider the case when a user gets an authorized access into the system, but performs some form of unauthorized use. For example, a specific user's guaranteed access to a local database does not mean that he has the authority to modify or delete any contents of this database. So, if he manages to defeat the authorization mechanisms or the control access enforcement, then he may perform an unauthorized use. This unauthorized use of the system and its components has to be prevented by designing and then implementing powerful methods to control the access. That means that the system administrator must configure the access control correctly and properly, and set a well-defined policy for the access control. The attacks and the countermeasure come in our taxonomy under Authorization Internet Access Control with Defeating Mechanisms and Policies **IAC-DMP**. On the other hand, a user may practice unauthorized use but at the same time he/she is a legitimate user. This approach, in our taxonomy, negates the naïve assumption made by Howard's taxonomy, "Because I felt that it was more important to emphasize the *unauthorized* nature of an attacker's activities, I chose to use the first pair of terms (*unauthorized access* and *unauthorized use*), but it should be understood that *unauthorized use* implies *authorized access*. In addition, it should be understood that *unauthorized access* implies that this access will result in an *unauthorized use*." [Howard, 1998]

Another utilization of our taxonomy is identifying different countermeasures based on the objective of the attack. For example, if the attacker succeeds to exploit communication protocol vulnerabilities such as IP address theft, then he may tamper with availability, data integrity, or any other security service. At this level of attack, all the necessary prevention countermeasures must be taken in order to stop the attacker from achieving his possible goals. To protect confidentiality, the private files and information have to be encrypted, so the attacker will fail to have a plain text of what he/she is looking for. This protection process may guard any sensitive data or information against

theft, but still, there is possibility that this information will be corrupted. Therefore, to protect the integrity, we need to continuously backup such information. This extent and link connection between the attacks and the countermeasures will give the researchers a powerful tool to conduct a focused and targeted kind of research. As explained, this taxonomy not only categorizes the security attacks, but also provides a means to identify the attacks and associate each attack with its proper countermeasures. For example, if we take a close look at the cell **C-PSI,** which depicts the case when the confidentiality as a security service is violated by penetrating the system or the individual's privacy class of attacks, such, the recommendation, through this taxonomy, to the user or system administrator is to use the properly established authentication system to eliminate or to reduce the risk of this attack. Under such attacks, a system can be monitored very carefully to detect any presence or attempt to retrieve the original text from the system, or any other unauthorized files transmission or malicious software. Moreover, user commands can be logged, and the resulting log is used to identify any attack on the system, and then to investigate the system's performance during and after the attack.

Every cell in the matrix (table 9) has the potential to be expanded for further research and investigation. Taking each cell in our matrix taxonomy would articulate an organized and detailed record of actual and potential threats with well-defined countermeasures. We are motivated by the idea of encouraging others to articulate a real and potential Internet security attacks. Our purpose is to identify the attacks and their class of category. Researchers should understand the concept and approach of our taxonomy, utilizing it to build a stronger security system. Every cell can, therefore, be used to map the associated security attacks to the security services designed to combat that attack.

From the above tables and the matrix taxonomy, we claim that our new classification of the Internet security attacks is exclusive. The approach toward an exhaustiveness of our taxonomy has been shown through some selective examples; more security attack examples can be subjected or processed through our taxonomy to prove our approach even further. The following subsection is the list and description of the carefully selected security attacks and countermeasures.

4.4.1 Attacks and Countermeasures Identifications

4.4.1.1 List of Security Attacks and Countermeasures.

NO.	Security Attack (SA)	Security Countermeasure (SCM)
1	**Passwords File Theft**	**Hashed Passwords File**
2	**Off-line Password Search**	**Forced Lengthy Trials**
3	**Trojan Horse/ Trojan login**	**Firewall**
4	**On-line Password Guessing**	**Audit Bad Passwords**
5	**Password Sniffing**	**Encrypted Password**
6	**Reply Hashed Password**	**One-time Password**
7	**Eavesdrops**	**Repeated Challenge Response**
8	**Weak Encryption Procedures**	**Suitable Encryption Procedures**
9	**Retrieve Original Plaintext**	**Overwrite Original Plaintext**
10	**Forged Biometrics**	**Detailed Sensing**
11	**Defeating matching mechanisms**	**Minimizing matching score.**
12	**Biometrics Interception**	**Biometrics Data Encryption**
13	**Biometrics Signature Replay**	**Authenticate the Biometrics Signature**
14	**Source Address Forgery**	**TCP Synchronization**
15	**IP Address Theft**	**GPS Location Authentication**
16	**TCP Splicing**	**Message Authentication**
17	**Synchronization Flooding**	**Connection Management**
18	**Distributed Denial of Service**	**IP Address Forgery Filtering**
19	**IP Spoofing**	**Unpredictable TCP Sequencing**
20	**IP Hijacking**	**Integrity of the Host OS**

NO.	Security Attack (SA)	Security Countermeasure (SCM)
21	**Denial of Service**	**Observing and Restrict Connection to the System;**
22	**OS Substitution**	**BIOS Password**
23	**Magnetic Retrieval**	**Multi Step Overwrite**
24	**Extract PIN**	**PIN Incorporated into Base Secret**
25	**Copy the Recovery Files**	**Encryption and Access Restriction**
26	**Forge Authorization Privilege**	**Encrypted Access Connections**
27	**Converts Reject into Accept**	**Keyed Hash Incorporating**
28	**Public Key Forgeries**	**Public Key Certificate**
29	**Substitute Certificate**	**Validate Certificate's Host Name**
30	**Sniffing a Private Key**	**Public Key on Smart Card**
31	**Steal a Private Key Backup**	**Private Key Created on Smart Card**
32	**Direct Communication Attack**	**Observing and Restrict Connection to the System**
33	**Back Doors**	**Eliminate Back Doors**
34	**Modifying Intercepted Message**	**Cryptography System**
35	**Social Engineering**	**Privacy and Unpredictable Password**
36	**Unauthorized Delete of Data**	**Limited Access**
37	**Time Bombs**	**Firewalls and/or Anti-virus**
38	**Buffer Overrun**	**Server Encapsulation**
39	**Mutual Trust**	**Firewalls and Enforce Access Control**
40	**Biometrics Sensor Disorientation**	**Check and Maintain Biometrics Sensor**
41	**Steal the Biometrics Token**	**Backup Emergency Processes**

Table 16: Security Attacks vs Security Countermeasures

4.4.1.2 Description of the Security Attacks and Countermeasures

We are presenting some of the potential security attacks and the security countermeasures associated with them, which are going to be classified according to where they fit in the above classification matrix.

SA-1 Passwords File Theft

Instead of targeting the individual password, the hackers can target the complete list of passwords. The main idea is that the attacker tries to find a weak point in the password system. This weak point might be the password file itself, which is a very valuable because it holds the passwords of all users.

SCM-1 Hashed Passwords File

To avoid the passwords file theft attack, the notion of encrypting the passwords has been introduced using a one-way hash function that would encrypt the passwords in an irreversible way. The procedure is to convert a plaintext password into an irreversible ciphertext that attackers cannot easily convert back into the password's plaintext. So, any attacker who might get an unauthorized access to the passwords file will not be able to read it. This technique will add more protection to the passwords file.

SA-2 Off-line Password Search

To defeat the protection of hashed passwords file, the attacker may try to generate the entire legal password and compare the resulting hash against those in the passwords file. This kind of attack can succeed only if there is a weakness in the encryption algorithm being used to hash the passwords. And the attack can be successful if there is great improvement in the computing speed.

SCM-2 Forced Lengthy Trials

The major defence against off-line password search is to make the password a very lengthy trial. In this case we can build the system to make each trail-and-error attempt take as long as possible by using, for example by forcing long passwords. Using a highly sophisticated encryption algorithm to hash the password is going to complicate

the search through the trail and error attempt, which is required because of the powerful technology being used in the computing systems.

SA-3 Trojan horse/ Trojan login

Trojan horse is malicious software that can cover itself behind something different. For example, one may download what appears to be a movie file, but it end up a dangerous program that can erase the hard disk, or send the credit card numbers and passwords to another party. Thus, Trojan horse attacks present one of a serious violation to web security. A very special case of Trojan horse is a Trojan login program, which is a program that tricks people into revealing their passwords. The only difference between Trojan horse and Trojan login is that, in the Trojan login attack, the user dose not need to run the excitable file himself, instead the attacker will do it to deceive the victim. When the attacker starts up the Trojan login attack, the victim can see perfect mimics to the login program that has been used on his system [Pelaez, 1991].

SCM-3 Firewall

It is pretty hard to detect and avoid Trojan horse attack, but one common security measure being used today is the Firewall. The firewall is blockade that being built to separate between an internal network which some how assumed to be secure and trusted, and another external (inter)network, such as the Internet, which is assumed not to be secure and trusted [Rolf, 1997], and [Preetham, 2002]. "Because the effect of a computer virus or Trojan Horse program cannot reliably be detected in advance of its action, it is vitally important to use a defensive mechanism that can prevent or at least detect any change to a program or data object, but which cannot itself be subverted, in order to prevent the further spread of the virus through the system." [Jueneman, 1998], for more on Virus attacks see [Lawton, 2002].

SA- 4 On-line Password Guessing

Some attackers may go after a passwords file, other attackers may use the on-line password guessing, in which they exploit human nature to try to guess what passwords people use. In a typical case, the attacker develops a list of possible passwords and makes

successive attempts to log on in to the web using the different passwords. Other attackers may try to prepare a list of most likely passwords depending on their knowledge of some social information [Yun, 1995].

SCM- 4 Audit Passwords

Guessing attacks can succeed if people are careless about password selection and if trial and error attacks proceed without detection. The first defence against such attacks is to keep an audit trail of attempts to log on to the system. An audit trail is a record of significant events within the system and is a common feature in modern computing systems and can be used to show when a user account is the target of an attack. In fact, same defence can be used against precompiled dictionary attacks or dynamic dictionary attacks [Steven, 1997].

SA- 5 Password Sniffing

In password sniffing, the attacker tries to intercept a copy of the secret password as it travels from its owner to the authentication mechanism. In general, any other computer on the web can read a message that has been transmitted across the web. The hackers, usually, have program that can read the messages which is not intended for them. So, if any one just log in and send his password in plaintext to the authenticator, the attackers can sniff it and use it later.

SCM- 5 Encrypted Passwords

Sniffing makes authentication on the web much trickier than on small private networks and LANs. An obvious solution is to encrypt the password, which means we need to transform the password's plaintext into a non-textual form that the attacker can't transform back into text. If the password is not in a readable textual form, then the attacker cannot type it in when reuse it to log on as the victim.
See as well, SCM- One-time Password.

SA-6 Reply Hashed Password

We can compute a one-way hash of the password and transmit that instead. This does not require a shared secret key. If the attacker intercepts the hashed password, he cannot turn it back into the original textual password, and he cannot use the hashed password directly to log in.

However, the hashed password opens us to a replay attack; in which the attacker can modify his system to simply provide an encoded password when he tries to masquerade as someone else. If the server simply expects a hashed password, then the attacker can simply replay the victim's hashed password. As a result, the server will not be able to tell if the password was generated from the original textual version or was simply replayed. The hashed password is an equivalent password that provides only a modest amount of security over using the plain-text password itself.

SCM-6 One-time Password

As a defence against the replay hashed password attack, we can use one-time password. This means that the service expects a new password each time the person logs on. The protocol used by the particular one-time password mechanism establishes how to generate the correct password. "One-time passwords (OTPs) have the advantage over regular passwords in that they protect legitimate users from replay attacks by generating a different password for each time of authentication" [Soh, 2003] Obviously, the one-time password has some problems, for example it is hard to remember a single password each time you try to login.

SA-7 Eavesdrops

In this attack, the attacker eavesdrops on someone who is trying to log on to a server with a one-time password. When the person sends the one-time password, the attacker intercepts the password, interrupts the communications path between the person and the server, and then uses the password himself to log on. The victim will most likely assume the problem was caused by network errors and is unlikely to suspect that an attack occurred. The person may try to log on again, and the server may accept this

attempt if people are allowed to be logged on more than once at a given time. If the server rejects the attempt, the user might still not suspect any attack happened.

Meanwhile, the attacker is successfully masquerading as the legitimate user. This is a very sophisticated attack, because the attacker must have control of the right portion of the network to be able to monitor the victim's traffic and interrupt the communications path to the server when necessary.

SCM-7 Repeated Challenge Response

One major defence against interception and replay attack is by using a challenge response, which is associated with a particular host or connection on the network, since it is supposed to represent the attempt to establish an authenticated session. If the same user attempts to complete the process of logging on from a different host, then the server is going to issue a new challenge response. Thus, the attacker cannot intercept a response and use it to log on from a different site or connection. This security measure does not resist a more active attack, such as IP address theft.

SA- 8 Weak Encryption Procedures

In the weak encryption attack, the attacker tries to find or to build a procedure that uses a shortcut to crack a weak encryption procedure. This part is going to be expanded on in more details when we talk about the attacks against different kind of encryption algorithms.

SCM- 8 Suitable Encryption Procedures

The solution is to replace the weak encryption with a well establish and more suitable one. This part is going to be expanded on in more details when we talk about the attacks against different kind of encryption algorithms and the security countermeasures used against them.

SA-9 Retrieve Original Plaintext

File encryption does not reliably protect data, if the encrypted file uses different disk space from the plaintext file. The attacker can simply retrieve a plaintext copy of the file if the application does not take special steps to erase the plaintext.. This strategy can work well on web systems, if the attacker targeting the hard drive or the database, but the attacker may be able to retrieve the deleted plaintext file through undelete option available in the system, or being installed as a malicious program.

SCM-9 Overwrite Original Plaintext

The defence against retrieve original plaintext attacks is to use an effective file encryption program to overwrite the plaintext file before deleting it. In most cases, it is enough to overwrite the original data with any other data. This prevents an attacker from retrieving the deleted file, copying it to another location, and then examining its contents.

SA-10 Forged Biometrics

The biometrics security services are being used in web authentication, e.g. the E-Commerce application. The biometrics reader is an obvious point of attack on a biometrics system. If the reader simply collects a digital image from a sensor and passes it on, then it can masquerade as someone by feeding the sensor the same data that the victim would provide. For example, in fingerprint systems; we produce an image of the victim's fingerprint and present that to the system instead of the victim's finger.

SCM-10 Detailed Sensing

The basic countermeasure for forged biometrics attack is to use more sophisticated sensors that measure additional properties of the physical entity. This is why many fingerprint readers try to detect the print in three dimensions, so that the ridges must actually be separate from the troughs to appear in the biometrics reading. Some fingerprint readers' sense skin capacitance or temperature in order to spot fakes. See as well SCM- Changing Behaviour

SA- 11 Defeating Matching Mechanisms

Most commercial fingerprint authentication systems use a minutiae-based system. This attack method inputs synthetic minutiae sets in the course of trying to defeat the matching system. The results returned by the matcher and the characteristics of the minutiae sets are used to generate a minutia set that exhibits high similarity to the information available in the database and hence results in a high matching score, thus achieving positive identification and defeating the system [Umut, 2003].

SCM- 11 Minimizing matching score

To counter such an attack, we need to improve the matching mechanism such only inputs that highly match will be authenticated. This is achieved by improving the technology behind the matching mechanism.

SA-12 Biometrics Disclosure

In this attack, an attacker may expose a person's biometrics data, while this data is being transmitted or stored, then the attacker might use it to monitor that person's private activities. In fact the biometrics systems are more likely to be accepted by users if the system takes steps to preserve the privacy of people's biometrics information. But with this biometrics disclosure attack, some biometric information is going to be revealed and the individual privacy is going to be violated.

SCM-12 Biometrics Data Encryption

The major defence against biometrics disclosure is to encrypt the biometrics data before transmitting it. For example, some models of fingerprint readers provide encryption to protect the fingerprint data as it travels from the reader to the computer. The biometrics database has to be hashed or encrypted as well.

SA-13 Replicating the Biometrics Signature

The attacker can replicate the biometrics signature of the victim if he manages to have a digitized copy of the victim's biometrics signature. The main idea is that each individual has a unique biometrics entity, so, he/she can uniquely be identified. But in case that the attacker succeed to reproduce or replicate a digitized copy of the victim's biometrics feature or, somehow, constructs a biometrics signature that matches it, and then transmits the signature to masquerade as the victim.

SCM-13 Authenticate the Biometrics Signature

To avoid the replicate biometrics signature attack we have to authenticate the source of the signature. Generally the computer cannot tell if someone is forging some else biometrics signature, except if we implement an authentication mechanism to authenticate the biometrics authentication data. To do that we need to install secret information in the biometrics reader and we can use the secret in conjunction between biometrics signature and a cryptographic integrity check. In case an attacker tries to forge a biometrics signature, its integrity check will fail, because the integrity check depends on the secret information. The integrity check will also detect attempts to modify a legitimate biometrics signature.

SA-14 Source Address Forgery

Each Internet message contains two numerical addresses: the source of the message and the destination of the message. If an attacker wishes to forge the source address, which means he is going to send an Internet message that appears to be from somewhere else, the attacker simply constructs the message so that it contains the desired source address. In some systems, this is simply a matter of changing the host's Internet address in the protocol stack's configuration information. Another approach is for the attacker to write special software to construct Internet messages and feed the false messages directly to the network device driver, bypassing the protocol stack entirely.

SCM-14 TCP Synchronization

The defence against the source address forgery comes in two parts. First defence comes from the design of the Internet itself. In another words, the Internet is designed in such away that no two hosts in the network have exactly the same Internet address. Second, is the synchronization protocol used to establish connections under the Internet's Transmission Control Protocol (TCP). TCP is the protocol used to establish a reliable connection for exchanging sequential data between two hosts. TCP carries a lot of important Internet traffic such as the Web connections.

See as well, SCM-Site Forgery Filtering, Message authentication, Secure RPC, and GPS Location Authentication.

SA-15 IP Address Theft

In IP address theft, the attacker tries to take over another host IP address. If he succeeds the attacker will masquerade as another and take over all traffic specified for that IP address. To be clearer, the attacker could, update the IP address used by the host protocol stack, and act as if this is the host's legitimate address, even though the address belongs to some other host.

SCM-15 GPS Location Authentication

GPS is a worldwide radio-navigation system formed from a network of satellites and ground stations that serve as reference points from which to calculate a person's geographical position to within a matter of meters [Braun, 2002]. The simplest thing can be done is to Transmit GPS signals to authenticate one's location at any given time.

SA-16 TCP Splicing

In TCP splicing, the attacker starts by monitoring the victim's traffic and collecting any information he might need to take over the victim's existing connections. The attacker has to monitor TCP sequence numbers so that he can continue using the connection. Once he has the necessary information, attacker breaks the network connection leading to the victim's machine and simultaneously announces that his computer owns that victim's Internet address. In fact, attacker splices the TCP

connections and reconnects them to his own computer. Once complete, the entire network treats the attacker's computer as if it is really victim's computer.

SCM-16 Message Authentication

To defence against TCP slicing or source address forgery we can use a source ID with any message that will be transmitted, and we should add a base secret to be associated with each source ID. Whenever the source wants to send a message, it attaches to the text of the message its source ID and its base secret, to a one-way hash function. The result of the hash is then added to message being sent. At the receiving side, the recipient will repeat the computation and compares the resulting hash with the one included in the message. If an attacker modified data in the message, then there will be no match between the original one and the one that has been computed by the recipient. Therefore, the attacker cannot forge a message because the message will not pass the authentication process.

SA-17 Abuse by administrator

Given the unlimited privileges of access that administrators have, there exists the risk of an administrator with malicious intentions abusing his privileges to gain access to a user's account for example and inflict some damage on the information and its availability.

SCM-17 Least Privilege

Keeping a log file that records individual transactions of administrators reduces the risk of administrators abusing the system. Although administrators are often too occupied to indulge in a profitable embezzlement while at the same cover their tracks, this record keeping is still essential as it provides accountability of actions.

SA-18 Distributed Denial of Service

DDoS attack is a more serious form of DoS attack, it relies on special attack programs that the attacker must install on numerous hosts and invoke remotely. Thus, the attacker must first attack and subvert a large number of computers on the Internet, and

use these hosts as the stage for mounting the DDoS attack on the real target. A strong attack might involve dozens or even hundreds of computers. To perform the attack, the attacker sends a message to each of the subverted hosts, telling them to start the DDoS attack. Then each host sends a flood of data at the targeted computer.

SCM-18 IP Address Forgery Filtering

The main defence for DDoS is to find the attacking computers one by one. This process is complicated because the attack traffic contains forged source IP addresses. Moreover, there is not necessarily any way of finding the actual attacker by looking at one of the subverted computers that is actually performing the attack. In this case we can use the IP trace back technique to filter the sources of all the messages, "to identify the source of any data sent across the Internet. Therefore, IP trace back can be used to trace the real sources of DDoS attack in which spoofed source IP addresses are usually used" [Wan, 2002]. More on defence against DDoS see [Belenky, 2003], see as well, SCM-Message Authentication

SA-19 IP Spoofing

In the IP spoofing, the attackers have developed a technique for forging the source location of an Internet packet for the purposes of tricking an rsh server. "In the IP spoofing attacks, the rlogin port of the target hosts was attacked because no authentication of the user is usually required when a connection is established. The next step is to start a shell (command processor) and invoke the attack code to control the account on the target host" [Hastings, 1996].

SCM-19 Unpredictable TCP Sequencing

The IP spoofing attack start by generating number of unsuccessful connections, in which the attacker can monitor and determine how the victim host generates its TCP sequence number. Therefore, and in order to defend against IP spoofing we should make

the sequence number hard to guess. Originally, the security implications of sequence numbers were only of interest to a few researchers. Following the IP spoofing attacks, unpredictable sequence numbers became a standard feature of TCP/IP protocol stacks, which we can consider as a major defence against IP spoofing.

SA-20 IP Hijacking

IP hijacking works by attacking a connection at one of its end-point; when a victim tries to connect to another host, the attacker steals the connection at the victim's end. The hijacking software is a special software package that gets installed as part of the operating system, and runs with the processor operating in privileged mode. The hijacking software then lets the attacker take over any connection that is currently set up.

SCM- 20 Integrity of the Host OS

It is hard to defend against IP hijacking. The attacker relies on software that becomes part of the operating system, and this can make it difficult to block, until we detect the presence of the hijacking software. This represents out principal defence to check the integrity of the host's operating system in order to detect the presence of suspicious software.

SA-21 Denial of Service

This kind of attack is an explicit attempt by attacker to prevent legitimate users from using the system. For example the attempt to flood the network with million of messages or by sending a millions of request to any specific site which can not handle, consequently the site will goes down. As a result the site will not be available to the legitimate users. More for more details of the DoS, DDoS, and other attacks scenarios see [Schetina, 2002].

SAM-21 Observe and Restrict Connection to the System

To prevent such an attack the user should disable any unused or unneeded network service; this will limit the capability of any attacker to take advantage of those

services to lunch a denial of service attack. Another precautionary countermeasure, users have to enable quota systems on the operating system. More importantly, users should observe the system performance and establish baselines for ordinary activity.

SA- 22 OS Substitution:

Most workstation allows someone to boot up an operating system from a different disk, like a CD-ROM or a diskette. This capability is provided for administrative or maintenance purposes. Attackers can use it to boot an operating system that is configured with access control protections disabled. Then the attackers can perform his objective damage; especially he can reconfigure the OS to disable its connection to rest of the system which affect the availability.

SCM- 22 BIOS Password

To protect against OS substitution attacks, workstations have to implement a BIOS password. This password is stored in the workstation's start-up configuration combined BIOS with instructions to always boot the workstation from a particular hard drive. The attackers can redirect the boot operation only if they know the password [BIOS].

SA-23 Magnetic Retrieval

Hard drives store data magnetically, and all data leaves a slight magnetic residue. A detailed laboratory analysis of magnetic patterns on a disk surface can retrieve data from a disk even if another layer of data has been written over it.

SCM- 23 Multi Step Overwrite

There is a relatively simple defence against magnetic retrieval attacks: apply a three-steps overwrite process to the data at least once. The process requires three binary data pattern: one random bit pattern (like 0011 0101), the complement of the pattern 91100 1010), and a pattern consisting of a mixture of bits from the two (like 1001 0111). Although this does not prevent all attacks, it is sufficient to make most attacks impractical.

SA- 24 Extract PIN

Using a token become very common in authentication systems and in most use there is a PIN number associated with the token. If a token is able to verify whether the PIN is correct all by itself without contacting a separate server, then there is enough information embedded in the software token for an attacker to recover the PIN.

SCM- 24 PIN Incorporated into Base Secret

As a countermeasure to the Extract PIN attack, PIN has to be incorporated into the base secret. This approach solves the problem faced by software tokens. The token does not even store a complete copy of the base secret. Instead, the actual base secret is constructed from a combination of the PIN and the token's partial base secret. If the attacker guesses the PIN incorrectly, he generates the wrong base secret value. The error does not appear until he actually tries to log on to the server, at which point the server detects and logs an incorrect one-time password. The attacker can not verify the PIN based on information within the software token, he must try to log on to the service, and that will produce a record of his attempt.

SA- 25 Copy the Recovery Files

Many NT systems provide an extra opportunity for attackers to steal the Security Account Manager (SAM) database; the recovery disk. NT systems allow administrators to maintain crucial system configuration files on a diskette called the recovery disk. One of the crucial files is, of course, the SAM database. Moreover, many systems maintain a copy of the recovery disk's files on line to simplify the process of keeping the disk up to date. Attackers have been able to recover the SAM database by stealing it from the recovery disk area of a domain controller. [Marcery, 1997]

SCM- 25 Encryption and Access Restriction.

There are two ways for preventing copy the recovery files attack. First, put access protection on the recovery disk files. Second, do not store the files on line except when actually constructing a recovery disk. In addition to that Microsoft has released a patch

to Windows NT to provide encryption of the SAM database. This mechanism, called the "System Key" of SYSKEY fix, uses a secret key to encrypt the SAM database. The computer's administrator must provide the key to the system somehow, and then the system decrypts the database and allows users to log on. [Microsoft]

SA- 26 Forge Authorization Privilege

This attack may happen when an attacker places himself between the agent and the authentication server. When any legitimate user tries to log on, attacker could intercept the authentication request on its way to the server and send back a forged authentication response that verifies user's logon attempt. If the agent can't detect the forgery, user will log on successfully.

SCM- 26 Encrypted Access Connections

Most computers using Windows network domain software automatically establish a secure channel to the domain control when they start up. The channel uses an encryption protocol to protect the information from attack. Windows uses this channel to carry pass-through authentication between server and their domain controller.

SA- 27 Converts Reject into Acceptance

Attacker may intercept packets that travel between the server and agent, modify them while they are in transmission and send them on their way. In particular, attacker may intercept the access reject messages sent in response to any client's access request and transform it into an access accept message. Attacker can do this by changing a single bit in the access reject message.

SCM- 27 Keyed Hash

See about the hash and keyed hash.

SA- 28 Public Key Forgeries

In this attack, attacker tries to convince people and/or software that his own public key is the one used by some important server, like bank's server. Attacker might be able

to profit even if he tricks only one or two people this way. For example, attacker could send a message to a victim pretending to be the bank, and give the victim the bank's public key when it's really his own key. If a victim then tries to send secret information to the bank, like password-protected message, attacker can masquerade as the bank.

SCM- 28 Public Key Certificate

The certificate is a data item containing a public key, the name of the key's owner, and a digital signature. We verify a certificate's authenticity by checking its digital signature. Certificates help us detect public key substitutions and man in the middle attacks. The attacker can not insert a bogus public key if we can verify the key's owner.

SA- 29 Substitute Certificate

An attacker might be the owner of a legitimate e-commerce site and have a legitimate public key certificate for the site's server. In such a case attacker can implement a second server that masquerades as any other server such as a bank serve and intercepts any victim's messages to the bank, and perform similar attack to the public key forgeries.

SCM- 29 Validate Certificate

In part the substitute certificate attack arises because the SSL security software operates at the software level, ignoring the data it carries, while the web server or other application software operates at a completely different level. The solution to the problem is for the application software itself to verify that the name in the certificate goes along with the messages being handled by the application. In the browser environment, Netscape solved this by checking the name in the certificate against the server being asked for in the HTTP. This is checking the name of the account holder on an electronic check against the name in the public key certificate. The check is legitimate if the certificate name matches the account's name. [SSL]

SA-30 Sniffing a Private Key

One of the risk faced by private keys stored in files is that subverted program might be able to sniff a user's private key while it is decrypted for use by the public key software.

SCM- 30 Public Key on Smart Card

In this approach, we use software on a workstation to generate the public key pair. Then we download the keys onto the smart card as part of its initialization. The card provides public key cryptographic functions without having to expose the private key. This eliminates the risk of having the private key disclosed to a sniffer on the workstation.

SA- 31 Stealing a Private Key Backup

After generating the Private Key, if smart card fails, user can create a new copy using his backed-up copy. This is particularly important if user apply his private key to protect stored files. Otherwise, user would lose the files' contents if his smart card failed. This strategy carries an obvious security risk: the more copies of a key we have, the greater is our risk that the key will somehow be leaked. Here we must trade off the benefit of a backup against the added risk of disclosure.

SCM- 31 Private Key Created on Smart Card

In this approach, a private key generation can be embedded on the smart card itself, and provide no way of disclosure of the private key. This addresses the risk of stealing a back-up private key as well as the risks associated with using the private key on a workstation. The only way to use the particular private key is to have possession of the smart card it resides on.

32- Direct Communication Attack

This attack represents one form of the Distributed Denial of Service Attack. In order for all the compromised machines and the master attacking machine to communicate with target they need to know each other identity. In fact the attacker IP address is going to be hard coded in the attacking code. Whenever any machine ready, it is going to declare that to the master attacker machine, so it can synchronize a massive attack against the target machine. The most crucial weakness in this kind of attack is that if the victim machine recognizes any of the compromised machines it can expose the whole aggressive machines that are lunching the attack [Jelena, 2002].

SCM- 32 Observe and Restrict Connection to the System

See the SCM- 21

SA- 33 Back Door

Back doors are features left behind by the system designer or system administrator. The attacker may use an access mechanism to gain an access to the system or any other Internet resources through a back door. Such an attack is hard to detect because the designer who puts it in place also make the access exempt from the usual audit logging feature of the system.

SCM- 33 Eliminate Back Door

The problem with back doors is that they are making more troubles than benefits. Most security experts find it challenging enough to design a system that meets the goal of preventing attacks. Attempting to add a safe, but hidden, back door would significantly increase the complexity of the system and increase the overall security risk. The safest strategy is to eliminate back doors in existing systems and to avoid products that contain them.

SA- 34 Modifying Intercepted Message

Message transmitted over the net can be intercepted. If the message is in plaintext, which means the data is readable, and then there is a real danger that the message will be altered or modified. For example, in the e-commerce, someone places a purchase order, pay for it, and provide delivery address. Attacker may intercept the message, modify delivery address, and pass the message again. This scenario is just simple the attack can be more serious and has a sever consequences.

SCM- 34 Cryptography Systems

To avoid any message forgery, encryption algorithm should be used. It is simply applying secret key to the plaintext to produce the ciphertext, and vies versa, applying a secret key on the ciphertext to produce the plaintext again, see chapter 2. We need strong encryption algorithm, At least the algorithm to be such that an attacker who knows the algorithm and can reach the cipher text should not be able to decrypt cipher text nor fined the key.

SA-35 Social Engineering

Using social relation to the people is one common way to get private information. Often, it is expected to find sensitive information in a personal notebook, or password written down on the documentation around the keyboard. Social engineering attack usually involves a telephone, emails, or any other social means.

SCM-35 Privacy and Unpredictable Password

Certain performance should not be allowed without strong authentication. For example, receiving a call at the work office and the caller claim he/she is the system administrator doesn't mean he is really the administrator; authentication is required in such a case, and so forth. Keeping secret and sensitive information in a secure and private place is another precautionary measure against social engineering attacks.

SA-36 Unauthorized delete/modify of Data

Applying authentication mechanism is not enough, in some case, to prevent aggressive attack. Legitimate user who has right identity can pass through the authentication process. But this legitimate user may perform an attack by practicing unauthorized use of the data with in the system. For example, an authentic person may get in to the system and delete some sensitive data, which he should not or has on right to do so.

SCM- 36 Limiting the Access

System administrator has to establish a policy that can monitor the performance of the user and enforce a pre defined security policy as well. After gaining an access into the system, user must be subject to limiting access policy, so he/she may have limited privilege according to his need and practice. System administrator or security personal has to enforce the access control mechanisms.

SA- 37 Time Bombs

A time or a logic bomb is a malicious code that may sty hidden and calm in the host system until a certain predefined time or event. When triggered, time bomb may cause different king of damage depending on the nature of the code itself. Time bomb may cause crash or damage to the operating system, delete files, etc.

SCM- 37 Firewalls and/or Anti-virus

See Firewalls

SA- 38 Buffer Overrun

The nature of this attack is to bypass security measures and run selected programs. It exploits any bugs or other vulnerability in the server and deceives it to download a code that can perform the objective of the attacker.

SCM-38 Server Encapsulation

The server software has to be design and operated with security in mind. For example, the server software has to be run in such away that restrict the access to the rest of the environment. So, if the attacker succeeded to breach through in to the server, he/she has a very limited access to the rest of the system, and can not run specific program that may cause any damage.

SA- 39 Mutual Trust

The attacker may use this method to masquerade as someone else, or to install malicious code on someone else machine. After break into one host, attacker may find another host that trust the first one and guarantee easy access where he can install malicious program.

SCM- 39 Firewalls and Enforce Access Control

See Firewalls, and Limited Access.

SA- 40 Biometrics Sensor Disorientation

Biometrics authentication processes rely, mainly, on the ability of the biometrics sensor to identify the biometrics objects. If there is any electronic damage, whither it is intentional or unintentional, that causes any disorientation in the functionality of the sensor this will led to authentication failure. Attacker may exploit such a vulnerability to break into the system and perform unauthorized use. For example, attacker may send massages on behalf of the legitimate user, which cause a repudiation dispute later. Disorientation

SCM- 40 Check and Maintain Biometrics Sensor

Security personal or system administrator has to regularly check the functionality of the biometrics sensor. For example, system administrator has to perform specific test on the sensor to insure that the authentication process is running well and any attack can

be detected and rejected. Hardware part of the biometrics sensor as well as the software has to be maintained and updated for the system to be always in perfect condition.

SA- 41 Steal the Biometrics Token

One of the problems with biometrics systems is the availability to the legitimate user when the user's biometrics gets injured. What security specialists had suggested is to save the biometrics information on a smart card for some availability purpose. In such a case, users have another problem if the smart card gets lost or stolen, legitimate user will no longer, at least for specific period of time, be able to log on into the system.

SCM- 41 Backup Emergency Processes

System administrator or service provider has to protect the right of a legitimate user to get an access into the system under a normal or emergency situation. In case of the biometrics token get stolen, system administrator must provide one time emergency access to the system by the legitimate user.

CHAPTER V
Conclusions & Future Work

Internet security comprises the operations that protect the information and the system that is processing the information by providing some basic security services like availability, integrity, authentication, confidentiality, and non-repudiation. These services may include the prevention mechanisms against any attacks or potential security attacks. Classifying the security attacks and services under specific categories to uniquely identify each one of them is what we call taxonomy. In general, taxonomy for security services must provide a historical revision of the security services and attacks in such a way that the software and system designers benefit in anticipating their systems flaws and vulnerabilities.

From the taxonomy, the designer can follow all the records under any specific classified group of attacks and services to analyze the weakness and vulnerabilities to become more knowledgeable, vigilant, and confident in building a better and more secure design. More specifically, the Internet security service taxonomy has to be detailed, comprehensive, and practical, so it can be utilized and used. A well done taxonomy can perform as framework to guide the security engineers and software scientists to fully understand where the system vulnerabilities are, how they perform under the attacks, what kind of attacks they should anticipate so that finally they can draw a complete picture of the security services and their techniques. The work of this thesis is an effort in producing a comprehensive taxonomy, which can address some of the flaws and shortcomings of the previous work in the literature.

Through our research in this thesis we have realized that the understanding of the security mechanisms, including both the services and the attacks, requires a deep understanding on how all of their components work. In addition to that, we should understand how the security components are interconnected and interrelated to each other. In other words, there is a need to decompose the security system into subsystems,

and try to analyze their entities, attributes of those entities, the interrelationship among them and the performance of those subsystems. Security performance of a system comes as a result of the performance of its subsystems and components. Vulnerabilities in any of the security components might be exploited by the attacker to launch his unauthorized access or use of the system.

In this thesis we have introduced a new method for driving Internet security service taxonomy, which shows a complete map between the security attacks and services. The method is based on the security attacks, the countermeasures, the security services, and the performance of the security system during any attacks. Our major objective is to provide sufficient understanding regarding the security service and Internet security. In that order, we have focused on satisfying the security requirements and analysis of the Internet security model, and at the same time, we took in our consideration the effects or potential effects that may be caused by the security attacks. Our taxonomy shows the vulnerabilities in a useful form, which may invoke prevention and provide suitable countermeasure. Furthermore, it shows common characteristics of related flaws to provide common security prevention approach.

The security taxonomy that we have produced describes the functional requirements of the Internet security mechanisms with respect to the performance of the whole Internet system. In fact, this taxonomy can be used for different purposes; it can be used to classify the security attacks, based on the classification categories that we have developed. Furthermore, this taxonomy can help classify the security services, link the security attacks to the security services, show the security system performance, and drive and assess best security policy.

The other contribution in this thesis is presented in the form of an extensive analysis of the most used Internet security algorithms. In particular, attacks on the systems and Internet applications running on them, and the security countermeasures developed to deal with these attacks from which we have concluded and derived our classification categories. Our taxonomy can serve as a reference or a handbook for all

people who are interested in conducting a study or research on the area of Internet security attack, Internet security services, and the security countermeasures associated with each attack with the current state of art. For example, when we classify the security attacks that use the Trojan horse as a tool to conduct the attack, the researchers can focus on developing a countermeasure associated with Trojan horse attacks and build on top of the techniques that have been categorized by our taxonomy.

To ensure robustness and reliability, our coverage has included most of the standard Internet attacks. We had to make sure that any security service algorithm or process avoids the most typical influence of the security attacks, which are: Disclosure, Traffic analysis, Masquerade, Content modification, Sequence modification, Timing modification, Source repudiation, and Destination repudiation. For example, the security concern while dealing with the Disclosure and Traffic analysis is message confidentiality. Measures dealing with Content modification, Sequence modification, and Timing modification have to deal with authenticating messages. Those measures are required while dealing with Destination repudiation and will have to utilize a combination of the use of digital signature and a protocol design.

This taxonomy provides a systematic and coordinated method to develop recommendations and guides for the computer users to follow for the purpose of maintaining and improving the Internet security and to gather and distribute useful information concerning Internet security. One more conclusion from our research is that the Internet security system has to be viable and adaptive to be able to cop with such a fast moving technologies. Furthermore, it has to be able to provide feasible performance in any different kind of Internet applications and platforms, even in a case where there is a sustain security risks and attacks such as the e-commerce.

Future Work

Definitely, the research on the Internet security system has to be expanded and formulated in more systematic approach. Internet technologies are moving very fast, as a result, new attacks and countermeasures, continuously, being introduced based on any new Internet technology. Therefore, there is always a need to update or introduce new classification categories of the Internet security attacks. In addition to that, future work may include a validation and verification of the taxonomy that we presented in this research. Such verification may be achieved by analyzing the performance of the taxonomy using newly introduced attacks and countermeasures, for example, researchers may apply new attacks which are developed based on a new emerging internet technology to our classification and identity where do they fit within our taxonomy. Our taxonomy has the potential to be optimized based on another dimension such as cost. The optimization and advantages of the research presented in this thesis will be in the volume and intensity of the future works and investigation that it motivates.

References

[Adam, 1995] Adam Shostack, May 1995, **An Overview of SHTTP**, **http://www.homeport.org/~adam/shttp.html**

[Alfred, 1996] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, **Handbook of applied cryptography**, 1996, USA, Page(s): 425-481, **http://www.cacr.math.uwaterloo.ca/hac/** .

[Anish, 1996] Anish Bhimani, **Securing the commercial Internet**, Communications of the ACM June 1996, Volume 39 Issue 6, Page(s): 29-35.

[Anonymous, 2001] Anonymous, **Maximum Security, A Hacker's Guide to Protecting your Internet Site and Network**, Sams Publishing, USA, 2001, Page(s): 36-68.

[Andrew, 2002] Andrew D. Gordon, Riccardo Pucella, **Session 2: secure Web services: Validating a Web service security abstraction by typing**, in the Proceedings of the 2002 ACM workshop on XML security, November 2002.

[Amoros, 1994] Edward G. Amoroso, **Fundamentals of Computer Security Technology**, Prentice-Hall PTR, Upper Saddle River, NJ, 1994, Page(s): 34.

[Belenky, 2003] **A** Belenky, N Ansari, **On IP Trace Back,** Communications Magazine, IEEE Volume: 41, Issue: 7, July 2003, Page(s):142 – 153.

[Bingyang, 2003] Bingyang Zhou; **An integrated web security system**, in the Proceedings of the 14th International Workshop on Database and Expert Systems Applications, 2003, 1-5 Sept, Page(s): 204 -208.

[Biometric Authentications, 2001] **Biometrics Authentications via the Internet**, **http://www.eyenetwatch.com/Biowebserver/fingerprint_authentication.htm** .

[BIOS, 2004] BIOS Password and Locked Hard Disk Recovery, http://www.pwcrack.com/bios.shtml

[Bishop, 1997] Bishop M., Cheung S. , Wee C. , **Threat from the Net**, Spectrum, IEEE , Volume: 34 , Issue: 8 , Aug. 1997, Page(s):56 – 63.

[Braun, 2002] Braun Martin, **A Secure Technology for Determining Client Computer, User and Location Authentication**, February 2002 **http://www.gamet.com/pdf/AgeLocationAuthentication.pdf**

[Caloyannides, 2000] Caloyannides, M.A.; **Encryption wars: early battles**, IEEE Spectrum, Volume: 37 Issue: 4, April 2000, Page(s): 37 -43.

[Carlton, 2001] Carlton R. Davis, **IPSec Securing VPNs**, the McGraw-Hill, USA, 2001, Page(s): 33-76.

[Curtin, 2002] Matt Curtin, **Developing Trust: Online Privacy and Security**, Apress, 2002, USA, Pages(s): 84-102.

[Dan, 2000] Dan Simon, Bernard Aboba, Tim Moore, **IEEE 802.11 Security and 802.1X**, March 2000, **http://www.ieee802.org/1/files/public/docs2000/8021xSecurity.PDF** .

[Edward, 2000] Edward W. Felten, Michael A. Schneider, **Timing attacks on Web privacy**, Proceedings of the 7th ACM conference on Computer and communications security, November 2000.

[Erik, 2002] Erik Schetina, Ken Green, and Jacob Carlson, **Internet Site Security**, Addison Wesley, USA, 2002, Page(s): 14-39.

[Garfinkel, 1996] Garfinkel, S.L.; **Public Key Cryptography,** IEEE Computer Volume: 29 Issue: 6, June 1996, Page(s): 101 –104.

[Garfinkel, 2002] Simon Garfinkel, Gene Spafford, **Web Security, Privacy and Commerce**, O'Reilly, USA, 2002, Page(s): 472-501.

[George, 2003] George Whitson, **Computer Security: theory, process and management**, The Journal of Computing in Small Colleges, June 2003, Volume.

[Hastings, 1996] Hastings, N.E.; McLean, P.A.; **TCP/IP Spoofing Fundamentals**, Computers and Communications, in the IEEE Conference Proceedings, 1996, Page(s): 218- 224.

[Horton, 2003] Mike Horton, Clinton Mugge, **Networks Security Portable Reference**, McGraw-Hill, California USA, 2003, Page(s) 119-139.

[Howard, 1998] John D. Howard, Ph.D. Thesis "**An Analysis of Security Incidents on the Internet**," 1998, **http://www.cert.org/research/JHThesis/Word6/**

[Householder, 2002] Householder, A.; Houle, K.; Dougherty, C.; **Computer attack trends challenge Internet security Computer**, Volume: 35, Issue: 4, April 2002, Page(s): 5 – 7.

[Introduction SSL, 1998] **Introduction to SSL, http://developer.netscape.com/docs/manuals/security/sslin/contents.htm**

[Irvine, 1999] Irvine, C.; Levin, T., **Toward a Taxonomy and Costing Method for Security Services,** in the Proceedings of the 15th Annual Computer Security Applications Conference, 1999., 6-10 Dec. 1999, Page(s): 183 –188.

[James, 2001] James Essinger, **Internet Trust and Security**, Addison-Wesley, Great Briton 2001, Page(s): 23-43.

[Java Card] **Java Card Special Interest Group;**
http://www.javacard.org/others/biometrics.htm

[Jelena, 2002] Jelena Mirkovic, Janice Martin and Peter Reiher, **A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms,** Computer Science Department University of California, Los AngelesTechnical report #020018, 2002

[Jerome, 2001] Jerome Burke, John McDonald, Todd Austin, **Architectural support for fast symmetric key cryptography**, in the Proceedings of the ninth international conference on Architectural support for programming languages and operating systems, Volume 34, 28 Issue 5, 2001, Page(s): 178-189.

[Jueneman, 1998] Jueneman R.; **Integrity controls for military and commercial applications**, Aerospace Computer Security Applications Conference, Dec. 1988, Page(s): 298 –322.

[Kitsos, 2002] P Kitsos, N Sklavos, O Koufopavlou, **An efficient implementation of the digital signature algorithm,** Electronics, Circuits and Systems, 2002. 9th International Conference on 15-18 Sept. 2002, Volume: 3, Page(s): 1151 –1154.

[Landwehr, 1994] Carl E. Landwehr, Alan R. Bull, John P. McDermott, and William S. Choi, "**A Taxonomy of Computer Security Flaws,**" ACM Computing Surveys, Vol. 26, No. 3, September, 1994, Page(s): 211-254.

[Lawton, 2002] Lawton, G. **Virus wars: Fewer attacks, new threats**, IEEE Computer, Volume: 35, Issue: 12, Dec 2002, Page(s): 22-24.

[Longstaff, 1998] John D. Howard, Thomas A. Longstaff, "**A Common Language for Computer Security Incidents",** October 1998.
http://www.cert.org/research/taxonomy_988667.pdf

[Lorrie, 1999] Lorrie Faith Cranor, **Internet privacy**, February 1999, Communications of the ACM, Volume 42 Issue 2.

[Lukatsky, 2003] Alex Lukatsky, **Intrusion Detection**, Alist Publishing, USA, 2003, Page(s) 29-78.

[MAC] Message Authentication Codes,
http://www.isg.rhul.ac.uk/msc/teaching/opt8/week6b.pdf .

[Marcery, 1997], Marcey Kelley, Wendall Mayson, **Windows NT Network Security Manager's Guide**, Department of Energy CIAC, December 1997.

[Michael, 1999] Michael K. Reiter, Stuart G. Stubblebine, **Authentication metric analysis and design,** ACM Transactions on Information and System Security (TISSEC), Volume 2 Issue 2, May 1999.

[Microsoft, 2003]
http://support.microsoft.com/default.aspx?scid=http://support.microsoft.com:80/support/kb/articles/q143/4/75.asp&NoWebContent=1

[Mitnick, 2002] Kevin D. Mitnick, William L. Simon, **the Art of Deception**, Wiley Publishing Inc, Indian USA, 2002, Page(s): 173-195.

[Moldovyan, 2003] Alexander Moldovyan, Nick Moldovyan, Doug Summerville, Vladimir Zima, **Protected Internet, Intranet and Virtual private Networks**, Alist Publishing, USA, 2003, Page(s): 95- 132.

[Neumann, 1998] Peter Neumann and Donald Parker, "**A Summary of Computer Misuse Techniques**," Proceedings of the 12th National Computer Security Conference, Page(s): 396--407, 1989.

[Niels, 1998] Niels Ferguson and Bruce Schneier, 1998 **A Cryptographic Evaluation of IPSec, http://www.counterpane.com/ipsec.html**

[NIST, 2001] National Institute of Standards and Technology (NIST), Special Publication, 800-33, "**Underlying Technical Models for Information, Technology Security**", December 2001, http://csrc.nist.gov/publications/nistpubs/800-33/sp800-33.pdf

[OSSTMM, 2003] Open Source Security Testing Methodology Manual Pitfalls (OSSTMM), Version 2.5, August 2003, http://ideahamster.org/projects/osstmm.htm

[Pelaez, 1991] Pelaez, C.E. ande Bowles, J. ,**Computer viruses**, SC; System Theory, 1991, IEEE, in the Proceedings of theTwenty-Third Southeastern Symposium, Page(s): 513-517.

[Preetham, 2002] V.V. Preetham, **Internet Security and Firewalls**, Premier Press Inc, USA, 2002 Page(s): 197-216.

[Perry, 2002] T. Perry and P. Wallich, **"Can Computer Crime Be Stopped?"**, IEEE , 2002, Spectrum, Vol. 21, No. 5.

[Richard, 1997] Richard E. Smith, **Internet Cryptography**, Addison Wesley Longman, Inc, Massachusetts, 1997, Page(s): 229-266.

[Rolf, 1997] Rolf Oppliger, **Internet security: firewalls and beyond,** Communications of the ACM, Volume 40 Issue 5, May 1997.

[Rubin, 1998] Rubin, A.D.; Geer, D.E., Jr.; **A Survey of Web Security**, Volume: 31 Issue: 9, Sept. 1998, Page(s): 34 -41.

[Satyanarayanan, 1989] M. Satyanarayanan, **Integrating security in a large distributed system**, ACM Transactions on Computer Systems (TOCS), August 1989, Volume 7 Issue 3, Page(s): 247-280.

[Schetina, 2002] Eric Schetina, Ken Green, Jacob Carlson, **Internet Site Security**, Addison-Wesley, USA and Canada, 2002, Page(s): 229-255.

[Scott, 2002] Scott, D.; Sharp, R., Developing **Secure Web applications,** Internet Computing, IEEE, Volume: 6 Issue: 6, Nov.-Dec. 2002, Page(s): 38 –45.

[Scott, 2003] Scott, D.; Sharp, R.; **Specifying and enforcing application-level Web security policies**, IEEE Transactions on Knowledge and Data Engineering, Volume: 15 Issue: 4, July-Aug. 2003, Page(s): 771 -783.

[Shai, 1999] Shai Halevi, Hugo Krawczyk, **Public-key cryptography and password protocols**, ACM Transactions on Information and System Security (TISSEC) August 1999, Volume 2 Issue 3, Page(s): 230-268.

[Shweta, 2002] Shweta Bhasin with NIIT, **Web Security Basics**, Premier Press, USA, 2002, Page(s): 231-289.

[SSL] **Secure Socket Layer,**
http://wp.netscape.com/security/techbriefs/ssl.html?cp=sciln

[Steve, 2001] Steve Burentt and Stephen Paine, **RSA Security's Official Guide to Cryptography**, the McGraw-Hill, USA, 2001 Page(s): 4-12.

[Stallings, 2003] William Stallings, **Cryptography and Network Security: Principles and Practices**, Prentice Hall, New Jersey USA, 2003, Page(s) 2-19.

[Steven, 1997] Steven M. Bellovin, Michael Merritt, **Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and**

password file compromise, in the Proceedings of the 1st ACM conference on Computer and communications security, December 1993, Page(s): 244-250.

[Soh, 2003] Soh, B.; Joy, A.; **A novel web security evaluation model for a one-time-password system**, in the Proceedings of the IEEE/WIC International Conference on Web Intelligence, 2003, Oct. 13-17, 2003, Page(s): 413 -416.

[Taxonomy, 2003]
http://whatis.techtarget.com/definition/0,,sid9_gci331416,00.html.

[Tulloch, 2003] Mitch Tulloch, **Encyclopedia of Security**, Microsoft Press, USA, 2003, Page(s): 302-327.

[Umut, 2003]Umut Uludag, Anil K. Jain, **Attacks on Biometric Systems: A Case Study in Fingerprints,** Department of Computer Science and Engineering,
Michigan State University; East Lansing, MI, USA, 2003.

[Wahab, 1999] Wahab, A.; Tan, E.C.; Heng, S.M., **Biometrics electronic purse,** Proceedings of the IEEE Region 10 Conference, Volume: 2, 15-17 Sept. 1999, Page(s): 958 –961.

[Walid, 2001] James B. D. Joshi, Walid G. Aref, Arif Ghafoor, Eugene H. Spafford, **Security Models for Web-based Applications**, February 2001, Communications of the ACM, Volume 44 Issue 2.

[Wan, 2002] Wan, K.K.K.; Chang, R.K.C **Engineering of a global defense infrastructure for DDoS attacks**; in the 10th IEEE International Conference, 27-30 Aug. 2002, Page(s): 419 –427.

[Yao-Wen, 2003] Yao-Wen Huang, Shih-Kun Huang, Tsung-Po Lin, Chung-Hung Tsai, **Data integrity: Web application security assessment by fault injection and behaviour monitoring**, in the Proceedings of the twelfth international conference on World Wide Web, May 2003.

[Yun, 1995] Yun Ding and Patrick Horster**, Undetectable on-line password guessing attacks,** ACM SIGOPS Operating Systems Review October 1995, Volume 29, Issue 4, Page(s): 77-86.

Appendix A: Abbreviations

Abbreviation	Description
WWW	World Wide Web
E-Commerce	Electronic Commerce
IP	Internet Protocol
IPSec	Internet Protocol Security
SSL	Secure Socket Layer
HTTP	Hyper Text Transfer Protocol
S-HTTP	Secure Hyper Text Transfer Protocol
CC	Common Criteria
FTP	File Transfer Protocol
NIST	National Institute for Standard and Technology
OSSTMM	Open Secure Security Testing Methodology Manual
IT	Information Technology
SQL	Structured Query Language
SA	Security Attack
SCM	Security Countermeasure
DoS	Denial of Service
DDoS	Distributed Denial of Service
SAM	Security Account Manager
MPSI	Manual Penetrating of the System and/or Individuals
DIR	Data Interception and Replaying
BPT	Biometrics and Physical Token

DMP	Defeating Mechanisms and Policies
MC	Malicious Code
DCS	Distributed Communication System
SRI	Stanford Research Institute