Abstract

In this paper I present a review of the different forms of network security in place in the world today. It is a tutorial type of paper and it especially deals with cryptographic algorithms, security protocols, authentication issues, end to end security solutions with a host of other network security issues. I compile these into a general purview of this topic and then I go in detail regarding the issues involved. I first focus on the state of Network security in the world today after explaining the need for network security. After highlighting these, I will be looking into the different types of Network security used. This part is quite an extensive coverage into the various forms of Network security. Then, I will be highlighting the problems still facing computer networks followed by the latest research done in the areas of Computer and Network Security.

1 Introduction

‘Data thieves strike Georgia Tech’ was the headline on CNETnews.com on March 31st of this year. Online intruders entered a server containing credit card numbers of about 57,000 patrons of a Georgia Institute of Technology arts and theater program and stole the information. The information was detected only last week and in this was in spite of the fact that this had been going on for the past 2 months. Another headline that captured world attention this year has been the attack of the virus slammer. The worm SQL Slammer which exploited a flaw in Microsoft's SQL Server database software, caused damage by rapidly replicating itself and clogging the pipelines of the global data network. The virus propagated so fast that it spread across the globe in less than 10 minutes cutting off web access in South Korea and shutting down certain Automated Teller Machines in the United States.

Now, in light of these recent events, especially since Network based attacks have been increasing in frequency and severity over the past several years, the need for a secure network have only increased. This paper, will thus be studying the challenges faced by the world in terms of Network security, followed by a study on the study on the forms of security present in the world today, which then looks into the latest research in the area of Network Security.

1.1 Organization of the Paper

The rest of the paper is organized as follows. In section 2, I deal with the network security issues. Section 3 will deal with the forms of network security in place in the world today. Section 4 will deal with the latest research in the area of Network Security and finally Section 5 will conclude the study.

2 Network Security Issues

In the first few decades of the existence of computers and related networks, computer networks were mainly used by university researchers for sending emails, sharing printers and a host of other related task that didn’t require any real monitoring. So security of the computer network wasn’t really of any importance. But now, with the advent of the Internet and millions surfing through cyberspace, either to shop, file tax returns or any other form of online transaction, Network security is very important. Network security is concerned with people trying to access services that they are not authorized to use. Network Security problems can be classified
mainly into four areas, secrecy, authentication, non-repudiation and integrity control. Secrecy has to do with keeping information away from the hands of unauthorized users. Authentication deals with determining who is trying to access the information. Non-repudiation deals with signatures which are used to make sure that the person is he/she who she claims to be. Now, the layer that is involved in network security, can be seen in the following way because each layer can play a part in Network Security. In the Data link layer, each packet can be encoded as they leave one machine and decoded as they enter another. In the Network layer, firewalls can be installed to keep packets in or to keep packets out. In the transport layer, entire connections can be encrypted that is, end to end, which means process to process. These solutions solve the secrecy problems but don’t solve the authentication and non-repudiation problems. These problems can be solved only in the application layer. Another problem faced by computer networks is the problem of flooding the network with packets leading to slowing down of the computer network. This leads to tremendous slowdown of the internet making certain sites inaccessible. In general, network security attacks can be divided into two, Passive and Active attacks. Passive attacks involve mainly eavesdropping, or monitoring of transmissions. Active attacks involve some modification of the data stream or the creation of a false stream and they can be mainly divided into four categories: masquerade, replay, modification of messages, and denial of service. A masquerade takes place when one entity pretends to be another entity. Replay involves passive capturing of a data unit and its subsequent retransmission to produce an unauthorized effect. Modification of messages simply mean that certain portions of a legitimate message are altered, or say, messages are delayed or reordered to produce an unauthorized effect. A Denial of service attack prevents or inhibits the normal use or management of communications facilities.

3 Network Security Types

We will at first look into Network security for secrecy problems. The passive attacks can be protected against by the use of cryptographic techniques where the information that is transmitted is changed into a format that can not be easily deciphered by the person eavesdropping.

3.1 Cryptography

Cryptography in a Computer Network deals with encrypting information using a particular cryptographic method and then sending it through a computer network and upon receipt of this information at the receiver, the information is decrypted and read. This is so that, anybody listening anywhere in between the two hosts between which the information is sent, will not be able to decipher the information being sent. The data that is sent is encrypted to a form called cipher text and this text is deciphered at the receiver. It works on the following principle, say if ‘P’ is data to be sent, then E(P) indicated encrypting of the message P, and D(E(P))=P, indicated decrypting of the encrypted message to give back the original text. The fundamental condition is that the receiver must know the encryption method with which the sender has encrypted the message that is ‘E’. Now, keeping the encryption method secret is not easy and can be compromised and its normally impractical to find a new encryption method for every situation so for this, what is known as a Key is used. This means, that, using a particular encryption method but a different Key every time, a different message is generated. In this way, only the secrecy of the key has to be maintained and in this way, different messages are generated by a particular method but with different Keys. The Key is taken as a short string that selects one of many potential encryptions. So in this case, whenever there is a compromise on the key, a new key has to be considered for encryption. The real secrecy is the key and the length of the key.

3.1.1 Historical Encryption Methods

Historical encryption methods can be divided into 2 categories: substitution ciphers and transposition ciphers.

3.1.1.1 Substitution Ciphers

In a substitution cipher, each of the letters or a group of letters is replaced by another letter or group of letters to disguise it. One of the oldest known ciphers that is there is the Caesar Cipher, attributed to Julius Caesar. In this method, a becomes D, b becomes E, c becomes F, d becomes G, and so on till z becomes C. A generalization of the Caesar cipher allows the cipher-text to be shifted by ‘k’ letters instead of 3 as in the Caesar cipher. Another improvement of
this is to have each of the symbols in the plaintext to be mapped onto some other letter. But this can be easily cracked if there is only a small amount of data, or say by counting the frequency of repetitions of the same text.

3.1.1.2 Transposition Ciphers

Transposition ciphers reorder the letters but do not disguise them like substitution ciphers. The cipher is keyed by a word or phrase not containing any repeated letters. The purpose of the key here is to number the columns. The plain text is written in horizontally, in rows and cipher text is read in columns starting with the column whose key letter is the lowest [1, 2].

3.1.1.3 One-Time Pads

To construct an unbreakable cipher, a method called one-time pad is used. Choosing a random string as a key, the plaintext is converted to a bit string. Then the compute the EXCLUSIVE OR of these two strings, bit by bit. The resulting cipher text can not be broken because every possible plaintext is an equally possible candidate. The cipher text gives no information at all and if there are large number of letters, each letter will occur equally often. The problem with this approach is that, the key can not be memorized so the sender and receiver has to have a written copy of it. So that can be compromised [1].

3.1.2 Modern Encryption methods

Modern cryptography uses the same basic ideas as traditional cryptography, that is transposition and substitution but the emphasis is different. Traditional cryptography used simple algorithms with no complexity involved but instead relied on long keys for their security. But modern cryptography, uses as complex algorithms as possible. Modern encryption algorithms can be divided into 2, secret key algorithms and public key algorithms.

3.1.2.1 Secret Key Algorithms

Secret key algorithms are those algorithms in which the key is kept a secret. The most famous secret key algorithms are DES and IDEA. The DES and IDEA are explained next.

3.1.2.1.1 DES

The DES (Data Encryption Standard) cipher algorithm developed by IBM has become the most popular secret key algorithm in the world. It is no longer secure in its original form but after modification, it can be still seen as very useful. Plaintext is encrypted in blocks of 64 bits, yielding 64 bits of cipher-text. The algorithm which is using a 56-bit key has 19 distinct stages. The first stage involves a key independent transposition on the 64 bit plaintext. The last stage is the exact inverse of this transposition. The stage prior to the last one exchanges the leftmost 32 bits with the rightmost 32 bits. The remaining 16 stages are functionally identical but are parameterized by different functions of the key. The algorithm has been designed to allow decryption to be done with the same key as encryption. The steps are just run in the reverse order. Each of the intermediate stages takes two 32-bit inputs and produces 32-bit outputs. The left output is simply a copy of the right input. The right output is the bitwise EXCLUSIVE OR of the left input and a function of the right input and the key for this stage. All the complexity lies in this function. In each of the 16 iterations, a different key is used. Before the algorithm starts, a 56-bit transposition is applied to the key. Just before each iteration, the key is partitioned into two 28-bit units, each of which is rotated left by a number of bits dependent on the iteration number. The key for a stage is derived from this rotated key by applying yet another 56-bit transposition to it. A different 48-bit subset of the 56 bits is extracted and permuted on each round [1, 2].

3.1.2.1.2 IDEA

DES, in its original form is insecure in spite of the fact that its still widely used especially in Automated Teller Machines throughout the world. So better block cipher algorithms were developed to overcome the limitations of DES. The most interesting and most important of these post DES block ciphers is IDEA (International Data Encryption Algorithm). The IDEA uses a 128 bit key so is much powerful than the DES. No currently known technique or machine can break the IDEA. The basic structure of the algorithm resembles DES in that 64-bit plaintext input blocks are mangled in a sequence of parameterized iterations to produce a 64-bit cipher text output blocks. Given the extensive bit mangling, eight iterations are more than
sufficient. As with all block ciphers, IDEA can be used in cipher feedback mode and other DES modes. The details of one iteration are as explained next. Three operations are used, all on unsigned 16-bit numbers. These operations are EXCLUSIVE OR, addition modulo $2^{16}$, and multiplication modulo $2^{16} + 1$. The operations have the property that no two pairs obey the associative or distributive property which makes breaking the code even more difficult. The 128 bit key is used to generate 52 sub-keys of 16 bits each, 6 for each of the eight iterations and 4 for the final transformation. Decryption uses the same algorithm as encryption only with different sub-keys [1].

### 3.1.2.2 Public Key Algorithms

The key distribution problem has always been the problem in cryptography. No matter how efficient a cryptographic algorithm was, if an intruder could steal a key, the algorithm was virtually worthless as it could come under attack. A new type of cryptosystem was developed to circumvent the problem faced by the use of Secret Keys and that was the system in which encryption and decryption keys were different. Also, it wasn’t possible to decipher the decryption key from the encryption key. Public key cryptography requires each user to have two keys: a public key, used by the entire world for encrypting messages to be sent to that user, and a private key, which the user needs for decrypting the messages. The former is called public key and the latter is called private key. The encryption algorithm and the public keys are made public and hence the name Public Key Algorithm.

#### 3.1.2.2.1 The Knapsack Algorithm

This algorithm was the first public key algorithm. It works on the following idea. The idea here is that someone owns a large number of objects, each with a different weight. The owner encodes the message by secretly selecting a subset of the objects and placing them in a knapsack. The total weight of the objects in the knapsack is made public, as is the list of all possible objects. The list of objects in the knapsack is kept secret. With certain additional restrictions, the problem of figuring out a possible list of objects with the given weight was thought to be computationally infeasible, and this formed the basis of the public key algorithm.

The knapsack algorithm was broken by the inventors of the RSA algorithm (section 3.1.2.2.2) and hence, is no longer considered secure and hence no longer used [1].

#### 3.1.2.2.2 The RSA Algorithm

The RSA algorithm gets its name from the initials of its 3 discoverers (Rivest, Shamir, Adleman). The RSA algorithm is a public key cryptographic algorithm which is based on some principles from number theory. The method works as follows. At first two large primes are chosen say ‘p’ and ‘q’ which are both typically greater than $10^{100}$. Then multiply ‘p’ and ‘q’ and equate that value to say ‘n’ and multiply ‘p-1’ and ‘q-1’ and equate that value to say ‘z’. Now, choose a number relatively prime to ‘z’ and call it ‘d’. Then find ‘e’ such that the product of ‘e’ and ‘d’ equals 1 mod z. After computing these parameters, encryption starts. At first the plaintext message ‘P’ is divided into blocks, so that each plaintext message, ‘P’, falls in the interval $0 < P < n$. This can be done by grouping the plaintext into blocks of k bits, where k is the largest integer for which $2^k < n$ is true. To encrypt a message, P, compute $C = P^e \mod n$. To decrypt C, compute $P = C^d \mod n$. To perform the encryption, one needs ‘e’ and ‘n’. To perform the decryption, one needs ‘d’ and ‘n’. Therefore the public key consists of the pair of (e, n) and the private key consists of (d, n). The security of the RSA algorithm is based on the difficulty of factoring large numbers [1, 2].

### 3.2 Authentication Protocols

Up till now, we were dealing with the first issue of network security, which was secrecy. Now, in this section we take a look at the second part which is Authentication. Authentication is the technique by which a process verifies that its communication partner is who he/she claims to be and not somebody else. Verifying the identity of a remote host requires complex protocols using cryptography. Authentication protects against active attacks.

#### 3.2.1 Message Authentication Code

One authentication technique involves the use of a secret key to generate a small block of data, known as message authentication code, that is appended to the message. This technique assumes that the two communicating parties, say
A and B, share a common secret key $K_{AB}$. When A has a message to send to B, it calculates the message authentication code as a function of the message and the key: $MAC_M = F(K_{AB}, M)$. The message plus code are transmitted to the intended recipient. The recipient performs the same calculations on the received message, using the same secret key, to generate a new message authentication code. The received code is compared to the calculated code to check for authenticity [2].

### 3.2.2 Authentication Using KDC

Establishing a secret key between two people who are communicating is no easy task. For instance, if one wants to talk to ‘n’ number of people this way, then one would need ‘n’ different keys. So this would urge a need for key management which will turn out to be a real burden. To circumvent this problem, a trusted Key Distribution Center (KDC) comes into the picture. In this model, each user has a single shared key with the KDC. The simplest known KDC authentication protocol is wide-mouth frog (so called after its inventor). The idea behind wide-mouth frog works as follows. A user say ‘A’ who wants to communicate with another user say ‘B’, picks a session key ‘$K_s$’ and tells the KDC that he/she wants to talks to the other user on the other side, in this case ‘B’ using this session key ‘$K_s$’. This message is encrypted with the secret key ‘$K_A$’ and tells the KDC that he/she wants to talks to the other user on the other side, in this case ‘B’ using this session key ‘$K_s$’. This message is encrypted with the secret key ‘$K_A$’ and tells the KDC that he/she wants to talks to the other user on the other side, in this case ‘B’ using this session key ‘$K_s$’.

### 3.2.3 Authentication Using Kerberos

Kerberos involves 2 additional servers in addition to the 2 clients or say the client and server that are communicating. The 2 additional servers are Authentication Server (AS) and Ticket Granting Server (TGS). An authentication server is similar to a KDC, in that it shares a secret password with every user. The TGS issues tickets that are used to convince the receiving client or server that the sender who has the ticket from the TGS is he/she whom they claim to be. It works like this. Suppose there are two users ‘A’ and ‘B’ who want to communicate with each other. ‘A’ sits down on an arbitrary workstation and types his/her name. The workstation sends his/her name to the AS in plaintext. The AS sends back a session key and a ticket, $K_{TGS} (A, K_s)$, intended for the TGS. These items are packaged together and encrypted using A’s secret key so that only ‘A’ can decrypt them. Only after this message comes from the AS, does the workstation ask for the password from ‘A’. The password is then used to generate $K_A$, in order to decrypt the message from the AS and to obtain the session key and the TGS ticket inside it. Now, ‘A’ tells the workstation that she wants to contact ‘B’. The work-station send a message to the TGS asking for a ticket to use with ‘B’. The key element in this request message is $K_{TGS} (A, K_s)$, which is encrypted with the TGS’s secret key and is used as proof that the sender is really ‘A’. The TGS responds by creating a session key, $K_{AB}$, for ‘A’ to use with ‘B’. The first is encrypted with only $K_s$ so that A can read it and the second is encrypted with only $K_B$ so that B can read it. Now, ‘A’ can send $K_{AB}$ to ‘B’ to establish a session with him/her. This exchange is time-stamped. After this series of exchanges, A can communicate with B using $K_{AB}$ [1].

### 3.2.4 Authentication Using Public-Key Cryptography

Mutual authentication can also be done using Public-Key Cryptography. Since the 2 users know each others public keys, all they have to do is agree on a session key which will serve as the key which they will use to encrypt their messages [1].

### 3.3 Digital Signatures

Digital signatures are used as a replacement for handwritten signatures on papers and documents which in this case would be digital documents. Digital Signatures can be seen divided into Secret key signatures, and public-key signatures.

#### 3.3.1 Secret key Signatures

In this case, a central authority is there that knows everything and whom everybody trusts. Each user then chooses a secret key and carries it by hand to the central authority’s office. When
say a user ‘A’ wants to communicate with a user say ‘B’, ‘A’ sends a message ‘P’ to the central authority’s office as $K_A(B, R_A, t, P)$. The central authority sees that it’s a message from ‘A’, decrypts it and sends the message to ‘B’ containing the plaintext of ‘A’’s message and also the signed message [1].

### 3.3.2 Public key Signatures

The biggest problem with secret key signatures is the use of secret keys in which there is a central authority and everybody has to listen to him/her for everything. Furthermore, the central authority can read all the messages. In this method, a plain text message say ‘P’ is encrypted using the private key of one of the communicating hosts involved and then this message is encrypted again using the public key of the receiver. When the message is received at the receiver, it at first decrypts the message using its private key and then, decrypts the message to generate the plaintext ‘P’ using the public key of the sender and in this way the original message can be verified and this indicates a public key signature [1].

### 3.4 Firewalls

Firewalls are just a modern adaptation of that old medieval standby: digging a deep moat around a castle. This design forces anyone entering or leaving the castle to pass over a single drawbridge, where they could be inspected by the I/O police. This analogy helps to explain the job of firewalls in computer networks. A company can have many local LAN’s but all traffic to and from the company is forced through an electronic drawbridge which in this case is a firewall. The firewall in this combination has two components: two routers that do packet filtering and an application gateway. The advantage of this design is that every packet must transit two filters and an application gateway to go in or out. No other route exists. Each packet filter is a standard router equipped with some extra functionality. This extra functionality allows every incoming or outgoing packet to be inspected. Packets meeting certain criterion are forwarded normally. Those that don’t pass the criterion are dropped. The packet filter on the inside checks outgoing packets and the one on the outside checks incoming packets. Packets crossing the first hurdle then go to the application gateway for further examination. The point of putting two packet filters on different LAN’s is to ensure that no packet gets in or out without having to pass through the application gateway and thereby ensuring that there is no path around it. Packet filters are driven by tables configured by the system administrator. These tables list sources and destinations that are acceptable, sources and destinations that are blocked and default rules about what to do with packets coming from or going to other machines [1, 8].

Firewalls can be generally classified as follows [8],

- packet filtering gateways or screening routers
- Stateful inspection firewalls
- application proxies
- guards
- personal firewalls

A packet filtering gateway or screening router is the simplest, and in some situations, the most effective type of firewall. A packet filtering gateway controls access to packets based on packet address (source or destination) or specific transport protocol type (such as HTTP web traffic). Packet filters do not "see inside" a packet; they block or accept packets solely on the basis of the IP addresses and ports.

Filtering firewalls work on packets one at a time, accepting or rejecting each packet and moving on to the next. They have no concept of "state" or "context" from one packet to the next. A Stateful inspection firewall maintains state information from one packet to another in the input stream.

An application proxy gateway, also called a bastion host, is a firewall that simulates the (proper) effects of an application so that the application will receive only requests to act properly. A proxy gateway is a two-headed device: It looks to the inside as if it is the outside (destination) connection, while to the outside it responds just as the insider would.

A guard is a sophisticated firewall. Like a proxy firewall, it receives protocol data units, interprets them, and passes through the same or different protocol data units that achieve either the same result or a modified result. The guard decides what services to perform on the user's behalf in accordance with its available knowledge, such as whatever it can reliably know of the (outside) user's identity, previous interactions, and so
forth. The degree of control a guard can provide is limited only by what is computable. But guards and proxy firewalls are similar enough that the distinction between them is sometimes fuzzy. That is, we can add functionality to a proxy firewall until it starts to look a lot like a guard.

A personal firewall is an application program that runs on a workstation to block unwanted traffic, usually from the network. A personal firewall can complement the work of a conventional firewall by screening the kind of data a single host will accept, or it can compensate for the lack of a regular firewall, as in a private DSL or cable modem connection. Just as a network firewall screens incoming and outgoing traffic for that network, a personal firewall screens traffic on a single workstation.

3.5 Intrusion Detection Systems

An Intrusion Detection system (IDS) [8] is a device, typically another separate computer, that monitors activity to identify malicious or suspicious events. An Intrusion Detection system is a sensor, like a smoke detector, that raises an alarm if specific things occur.

Intrusion Detection systems monitor users and system activity, auditing system configuration for vulnerabilities and miss-configurations, assessing the integrity of critical system and data files, recognizing known attack patterns in system activity, identifying abnormal activity through statistical analysis, managing audit trails and highlighting user violation of policy or normal activity correcting system configuration errors installing and operating traps to record information about intruders

The two general types of intrusion detection systems are signature based and heuristic. Signature-based intrusion detection systems perform simple pattern-matching and report situations that match a pattern corresponding to a known attack type. Heuristic intrusion detection systems, also known as anomaly based, build a model of acceptable behavior and flag exceptions to that model; for the future, the administrator can mark a flagged behavior as acceptable so that the heuristic IDS will now treat that previously unclassified behavior as acceptable.

Intrusion detection devices can be network based or host based. A network-based IDS is a stand-alone device attached to the network to monitor traffic throughout that network; a host-based IDS runs on a single workstation or client or host, to protect that one host.

An IDS is a network device (or, in the case of a host-based IDS, a program running on a network device). Any network device is potentially vulnerable to network attacks. To counter those problems, most IDSs run in Stealth mode, whereby an IDS has two network interfaces: one for the network (or network segment) being monitored and the other to generate alerts and perhaps other administrative needs. The IDS uses the monitored interface as input only; it never sends packets out through that interface. Often, the interface is configured so that the device has no published address through the monitored interface; that is, a router cannot route anything to that address directly, because the router does not know such a device exists. It is the perfect passive wiretap. If the IDS needs to generate an alert, it uses only the alarm interface on a completely separate control network.

4 Latest Research

Much work is being done to enhance the security of networks. Research by vendor companies will lead to more flexible and secure boxes, while more fundamental research will look into the fundamental problems of networking: authentication, access types, and authorizations. A particular problem of security in networks is one of speed: As the speed, capacity, bandwidth, and throughput of networks and network devices continue to increase, security devices have to keep pace, which is always a challenge.

A second security challenge with networks is ubiquity: As automobiles, cell phones, personal digital assistants, and even refrigerators become network enabled, they need security. The need for a firewall for a cell phone will become apparent the first time a cell phone is subject to a denial-of-service attack. Once again, security will be called upon to protect after a product is in use [8].

Now, let's take a look at the various areas of research that are pursued in the areas of Network Security.

4.1 Intrusion Detection Systems

Researchers have been working on intrusion detection systems for a long time, without achieving what could be called a major breakthrough. The usual line of research focuses
on what are called "Anomaly Detection Intrusion Detection Systems" (AD-IDS). In principle an AD-IDS "learns" what constitutes "normal" network traffic, developing sets of models that are updated over time. These models are then applied against new traffic, and traffic that doesn't match the model of "normal" is flagged as suspicious. AD-IDS are attractive conceptually, but they require training, and the sad reality of networking is that it's very hard to classify "normal" traffic. As networks get sufficiently large, the applications mix they carry becomes so complex that it looks effectively random. An attacker may even generate traffic to generate a distorted model of "normal" so that sooner or later, an attack may look "normal" and get past the IDS. If the IDS is conservative about what may constitute an attack, it will tend to generate large numbers of "false positives" – false alarms. There is still a great deal of research being done in AD-IDS, and promising new avenues include marrying security analysis with data mining and visualization techniques. Many commercial firms are producing a simpler, easier to operate form of IDS called "Misuse Detection Intrusion Detection Systems" (MD-IDS). The MD-IDS closely resembles a virus scanner attached to a network. Usually, it is programmed with signature sets that represent the types of connection and traffic that indicate a particular attack is in progress. MD-IDS strength is that it is fast, and doesn't generate false positives because it "understands" what attacks look like. The weakness of MD-IDS' is that, like virus scanners, they cannot detect something they don't know about [9].

4.2 Cryptography

Cryptographic algorithms have been developed and fine-tuned almost all the time. The latest in cryptographic algorithms have been the Master Key cryptosystems [10]. This is a new class of secret key cryptosystems in which an authorized third party possess a Master Key that allows recovery of the clear text without knowledge of the session key. Otherwise an MKCS operates and is used by ordinary users exactly as any secret-key cryptosystem is used. In particular, pairs of ordinary users must agree on a shared key before they can communicate. MKCSs should be secure against ordinary attacks, because knowledge of only the session keys or the master keys shouldn't allow recovery of clear text. MKCS is a less cumbersome alternative to key escrow when third party access is required, since the escrow occurs automatically when the cryptosystem is designed. Efficient MKCSs are roughly equivalent to public-key cryptosystems in which encryption is very fast and key-generation is slow.

Research at IBM in cryptography is such that there is a process to develop a Federal Information Processing Standard (FIPS) for Advanced Encryption Standard (AES) specifying an Advanced Encryption Algorithm (AEA) and has been initiated by the National Institute of Standards. It is intended that the AES will specify an unclassified, publicly-disclosed encryption algorithm available royalty-free worldwide, that is capable of protecting sensitive government information well into the next century. It is also hoped that this standard will be as widely accepted as the Data Encryption Standard (DES) in the private and public sectors. IBM has been actively involved in cryptography research for many years. The DES standard for symmetric key encryption using 56 bit keys was designed and proposed at IBM several years ago. IBM has now developed a new encryption algorithm using 128 bit keys called MARS and has submitted it to NIST as a possible candidate for the AES. They are also working on the cryptanalysis (especially of other ciphers submitted to AES) and on elliptic curves. Elliptic curves possess a algebraic group structure which is used both to design public key signature algorithms and to break other schemes [16].

Research at Stanford is focused now on Intrusion Tolerance via Threshold Cryptography (ITTC). The ITTC provides tools and an infrastructure for building intrusion tolerant applications. Rather than prevent intrusions or detect them after the fact, the ITTC system ensures that the compromise of a few system components does not compromise sensitive security information. To do so we protect cryptographic keys by distributing them across a few servers. The keys are never reconstructed at a single location. [11]

4.3 Firewalls

To address issues and problems associated with IPV4, the next generation of Internet Protocols and IPV6 (specifically the next version of IP) incorporate optional security headers [13]. Because the security headers provide the basis for robust authentication, integrity, and confidentiality, services deemed insecure with IPV4 could be quite secure with IPV6 (provided
that the security headers option is used). As a result the threat posed to a system using IPv6 should be significantly less than that posed to an IPv4 based system depending on the extent to which the security headers are used. IPv6's security headers correct some problems that current firewall technology cannot correct, such as session stealing, in which an attacker can take over an established connection. IPv6 security headers and related items have been defined, but how the headers will be used in conjunction with security gateways and other systems is still open to debate and experimentation. IPv6 security services could be used directly between hosts with no security gateway intervention, which would indicate that the security gateway may become involved only in those security functions that IPv6 does not handle. Another new technology in firewall implementation is the Distributed Firewall. Conventional firewalls rely on topology restrictions and controlled network entry points to enforce traffic filtering. Furthermore, a firewall cannot filter traffic it does not see, so, effectively, everyone on the protected side is trusted.

To address the shortcomings of firewalls while retaining their advantages, proposed the concept of a distributed firewall [15]. In distributed firewalls, security policy is defined centrally but enforced each individual network endpoint (hosts, routers, etc.). The system propagates the central policy to all endpoints. Policy distribution may take various forms. For example, it may be pushed directly to the end systems that have to enforce it, or it may be provided to the users in the form of credentials that they use when trying to communicate with the hosts, or it may be a combination both. The extent of mutual trust between endpoints is specified the policy. To implement a distributed firewall, three components are necessary: A language for expressing policies and resolving requests. In their simplest form, policies in a distributed firewall are functionally equivalent to packet filtering rules. However, it is desirable to use an extensible system (so other types of applications and security checks can be specified and enforced in the future). The language and resolution mechanism should also support credentials, for delegation of rights and authentication purposes. A mechanism for safely distributing security policies. This may be the IP key management protocol when possible, or some other protocol. The integrity of the policies transferred must be guaranteed, either through the communication protocol or as part of the policy object description. A mechanism that applies the security policy to incoming packets or connections, providing the enforcement part.

Traditionally, hosts have tended to assign relatively few network addresses to an interface for extended periods. Encouraged by the new abundance of addressing possibilities provided by IPv6, a new method, called Transient Addressing for Related Processes (TARP) [14], is used, whereby hosts temporarily employ and subsequently discard IPv6 addresses in servicing a client host’s network requests. The method provides certain security advantages and neatly fineses some well-known firewall problems caused by dynamic port negotiation used in a variety of application protocols.

In the simplest of inter-networked host models, a client or server host has a single network interface with a single network address identifying the host. Even under such an elementary set-up, firewalls have traditionally faced difficulty when confronted with application protocols needing to open secondary channels. Examples abound, most notably ftp, but also rsh, RealAudio, H.323, tftp and the X Window System. To operate with such popular applications, firewalls have been forced either to follow the application layer protocol and configure themselves appropriately or to keep open, sometimes unnecessarily, a range of ports. As an alternative to potentially complex, detailed, and often Stateful firewall interaction, a method using multiple network addresses per host to organize and simplify firewall decisions is used. Under this basic model, instead of trying to follow the unfolding application protocol details, the firewall makes an initial permissibility determination based on transport layer protocol and the endpoints’ ports and addresses. Assuming approval of the proposed transaction, the firewall subsequently permits all traffic between the approved address pairs, irrespective of port.

Research at Stanford is the development of a system that controls access to computer networks through publicly accessible LANs, enabling network administrators to authorize users either on a permanent or occasional basis. The system has been designed with minimal assumptions about the software and hardware required of users, and requires very little specialized equipment within the network infrastructure. This access control mechanism that they are working on right now is Secure Public Internet
Access Handler (SPINACH) [12]. In SPINACH, a self-configuring router controls per-user access from a public subnet to a private one, using Kerberos or a similar mechanism to authenticate users and provide an audit path before users are granted access. With the exception of one custom software component on the router, SPINACH uses only standard protocols and software and requires only minimal software on users' machines. The SPINACH system establishes a "prisonwall," controlling the flow of packets between those hosts connected to public ports and the rest of the building network. As opposed to a firewall, which protects machines inside a particular network from malicious users outside the network, the prisonwall protects machines outside one portion of a network by refusing to forward packets that come from unauthorized hosts within. As users within the prisonwall authenticate themselves and thus activate network access for their hosts, SPINACH maintains an audit trail so that the users can be held accountable for traffic they generate on the network.

5 Conclusion

Throughout this study, we have looked at the various aspects of network security and have got the opportunity to see the different issues in computer security, the various problems plaguing the various computer networks and finally the research undertaken to better the utilization and to minimize the potential harm in network traffic and a host of other issues. This entire overview and study of the various aspects of Network security, has been able to show the issues that Network security experts and systems and LAN administrators have to deal with and the issues of concern to them. Network security is a rapidly changing field, but there are many current works and some classics everyone should read. To know more about these and other Network Security topics, looking into the links provided in the References section and also, referring the reference books will give more detailed answers.

6 References


