SECURITY METHODS, SYSTEMS AND PRODUCTS FOR E-COMMERCE

CMPE296U
Instructor: Dr. Jerry Gao

A Survey Report

By

Anjali Shende
Date: 10/07/99
ABSTRACT

Electronic commerce is a new way of interacting, bartering, and transacting with people and businesses. It is a revolutionary form of industry that enables organizations to interact electronically with customers. It allows companies to be more flexible and efficient in their operations, and it is also helps build closer relationships with customers and suppliers alike. But more importantly, E-commerce can extend a company’s reach to global level. As in any other means of business, we cannot assume all players will abide by a code of moral conduct. The business is being performed online over an insecure medium is enough to entice criminal activity to the Internet.

The most significant challenge to E-commerce is security - both internal and external. Ensuring secure online transaction is vital to success of E-commerce. Consumers and businesses must feel confident that their information will be safe. A number of technological solutions have been introduced to assure end users and businesses of privacy and confidentiality in online transactions. This survey report examines the security concerns for users and businesses engaging in Internet based e-commerce. It also surveys the different security methods and system currently being used for the e-commerce security. A number of companies has developed different security products. These different companies and their product are studied and analyzed, and a comparison is made.
# Table of Contents

**Abstract** ...................................................................................................................... 1

1.0 Introduction .................................................................................................................. 4

1.1 Confidentiality ............................................................................................................. 5

1.2 Authentication ............................................................................................................. 6

1.3 Data Integrity .............................................................................................................. 7

2.0 Electronic Transactions and Security ............................................................................ 8

2.1 Basic Terminology ...................................................................................................... 8

2.2 Example of a Secure Electronic Transaction .............................................................. 8

2.2.1 The Traditional Check Writing Process ................................................................ 8

2.2.2 The Electronic Version ............................................................................................ 9

2.3 Privacy and Encryption .............................................................................................. 9

2.4 Public Key Cryptography for Delivering Symmetric Keys ......................................... 9

2.4.1 Digital Signature ..................................................................................................... 10

2.4.2 Key Management ................................................................................................. 13

3.0 Symmetric Key Algorithms ........................................................................................ 15

3.1 Symmetric or Secret Key ......................................................................................... 15

3.2 Algorithms ................................................................................................................ 15

3.2.1 DES: Data Encryption Standard ............................................................................. 15

3.2.2 Triple-DES ............................................................................................................ 15

3.2.3 IDEA ................................................................................................................... 16

3.2.4 AES .................................................................................................................... 17

3.2.5 RC2, RC4, and RC5 .............................................................................................. 17

3.2.6 SAFER ............................................................................................................... 18

3.2.7 Cryptographic Hash Functions ............................................................................ 18

4.0 Public Key Cryptography – Methods & Systems .......................................................... 19

4.1 Diffie-Hellman Key Exchange .................................................................................... 19

4.2 RSA: Rivest, Shamir and Adleman ............................................................................. 20

4.3 ElGamal (Discrete Logarithm Systems) ..................................................................... 20

4.4 Elliptic Curve Algorithms ........................................................................................ 21

4.5 Comparisons of Encryption Systems ....................................................................... 22

5.0 Products, Analysis, and Comparisons ...................................................................... 24

5.1 Cybersafe ................................................................................................................ 24

5.1.1 Trustbroker™ Security Server ............................................................................. 24
5.1.2 DEFENSOR® PRODUCT SUITE _______________________________________________25

5.2 ID2                                                                                       26
  5.2.1 ID2 CERTIFICATE MANAGER ________________________________________________________26
  5.2.2 ID2 PERSONAL ___________________________________________________________________26
  5.2.3 ID2 SECURE TRANSPORT __________________________________________________________27
  5.2.4 ID2 SERVANT____________________________________________________________________27
  5.2.5 ID2 GUARDIAN _________________________________________________________________28

5.3 BALTIMORE TECHNOLOGIES _____________________________________________________________28
  5.3.1 UNICERT V3.05 ___________________________________________________________________28
  5.3.2 W/SECURE, X/SECURE, PKI-PLUS ___________________________________________________________________29

5.4 XCERT INTERNATIONAL, INC. ___________________________________________________________29
  5.4.1 SENTRY CA, SENTRY RA, & WEB SENTRY _____________________________________________30
  5.4.2 XCERT DEVELOPMENT KIT (XDK) ___________________________________________________30

5.5 IBM (INTERNATIONAL BUSINESS MACHINES) _______________________________________________30
  5.5.1 SECUREWAY _________________________________________________________________30

5.6 CERTICOM _________________________________________________________________________31
  5.6.1 SSL PLUS 2.0 ____________________________________________________________________31
  5.6.2 SC-400 SERIES SMART CARD ______________________________________________________31

5.7 ENTRUST TECHNOLOGIES __________________________________________________________________31
  5.7.1 ENTRUST/EXPRESS __________________________________________________________________31
  5.7.2 ENTRUST/ACCESS, ENTRUST/VPN CONNECTOR __________________________________________32
  5.7.3 ENTRUST SECURITY (FOR SAP R/3) __________________________________________________32
  5.7.4 ENTRUST/PKI ________________________________________________________________33

5.8 RSA SECURITY ______________________________________________________________________33
  5.8.1 RSA AUTHENTICATION SUITE (RSA ACE/SERVER AND RSA SECURID) _________________33
  5.8.2 RSA KEON ______________________________________________________________________34

5.9 NETSCAPE ______________________________________________________________37
  5.9.1 NETSCAPE CERTIFICATE MANAGEMENT SYSTEM ______________________________________37
  5.9.2 NETSCAPE SECURITY SERVICES _________________________________________________38

5.10 VERISIGN ________________________________________________________________________38
  5.10.1 CHECK POINT VPN-1 & CHECK POINT SECUREMOTE ___________________________________39
  5.10.2 VERISIGN DIGITAL ID __________________________________________________________________39
  5.10.3 VERISIGN ONSITE (PKI) _________________________________________________________40

5.11 ENTEGRITY SOLUTIONS __________________________________________________________________40
  5.11.1 NOTARY ________________________________________________________________________40
  5.11.3 SDP – SECURITY DEVELOPMENT PLATFORM __________________________________________41
  5.11.4 ASSUREWEB ________________________________________________________________42

5.12 KERBEROS ________________________________________________________________________43

5.13 PRODUCT COMPARISON _______________________________________________________________45

REFERENCES ________________________________________________________46
SECURITY METHODS, SYSTEMS, AND PRODUCTS FOR ELECTRONIC COMMERCE

1.0 INTRODUCTION

Electronic commerce or E-commerce is simply business conducted over the Internet. The process of buying, selling or interacting with customers via Internet, smart cards, or other computer networks is referred to as electronic commerce. E-commerce allows companies to be more flexible and efficient in their operations, and it also helps build closer relationships with customers as well as the suppliers. And more importantly, E-commerce helps to easily extend a company’s reach to a global level. Though electronic commerce is still in its infancy, it is expected to become more extensively used throughout the world in not too distant a future.

When it comes to doing business (over the Internet or otherwise), privacy and protection of sensitive data are primary concerns. With more and more people becoming familiar with the Internet and its processes; and with the fact that you can do almost anything over the Internet, from talking to people half way across the world to buying merchandise and paying bills privacy and data security has become one of the top issues. In the virtual world like the Internet, it is hard to accept that no danger lurks around the corner or that everyone, who has a Web site is a genuine, honest person. We all know that among us, there are a few dishonest merchants, who want to break into your computer, steal your credit card information and any other personal information they can get their hands on. Protecting yourself against such elements becomes necessary. Threats to security has a thousand different faces – vandalism and sabotage on the Internet, breach of privacy or confidentiality, theft and fraud, violations of data integrity, denial of service attacks etc. It is not possible to list and do justice in this paper to the numerous ways security can be compromised.

Information security is a necessary underpinning for further advances in electronic business. Technologies such as session encryption, encryption of transported data, firewalls, virtual private networks, digital certificates, and authentication have all emerged as pieces of the solution. Each is designed to enhance some aspect of information security, either by restricting access to or preventing the interception of private data.

Four main issues in E-commerce security can be identified:

1. Web-client security – the security of the machine used by the customer to navigate on the web.
2. Web-server security – the security of the machine used by the merchants to display and sell their products
3. Operating system – how good your operating system is in preventing unauthorized people from accessing or tampering with your data
4. Data transport security – this relates to the secure exchange of data over the Internet.
Data transport security has generated a lot of attention and substantial effort is being invested both by the private companies like RSA and government agencies like NIST in its development. Participation by the highest government agencies underscores its importance not only for commerce but for its role in national security as well. This is indeed a double edged sword. Secure data transport enables merchants and customers to conduct their daily business without any fear of fraud or of losing privacy. However, the very same mechanism which makes it secure can be used by people intent on creating a mischief. It makes it very difficult for the federal agencies to intercept messages from such parties which would allow them to identify and tackle rogues. Such being the case, government has been in a quandary whether to allow use of powerful methods of data transport and whether to allow the export of such technologies. The reluctance to approve the use and export of security technology had given an edge to foreign companies where there are hardly any government restrictions. Only recently has the US government lifted these restrictions. It will, undoubtedly give a new impetus to research and development in this field.

This paper has data transport security as a topic of its focus. Ways of securing the operating system, use of Firewalls, and securing the client and server software will not be dealt with here.

Data transport security can be further classified into four categories:
1. Confidentiality
2. Authentication (server and client authentication) (whether you are who you say you are)
3. Data Integrity
4. Non-repudiation (cryptographic receipts are created so that the author of a message cannot deny sending that message.)

Data need to be protected so that wrong people will not alter it (Data Integrity). When there is no data integrity it may become possible for hackers to steal data from your web-site or for the graffiti artists to just deface it.

It has to be made sure that certain kinds of data are protected so that only authorized people can access it (User Authentication). When there are problems with user authentication, licensed data might become accessible to someone who shouldn’t have access to it. Grades of one student might be read by another; or a professor’s exam answer sheet might be read by the students instead of just the TA. Also, due care has to be taken to ensure that sensitive data that a client or a customer sends to a particular server – e.g. a social security or a credit card number – goes to the right server (Server Authentication). When a server authentication is broken a user might end of sending his credit card information to the wrong server; or personal information that the user doesn’t want the world to know may be given to the wrong people, and so on.

1.1 CONFIDENTIALITY

From e-mail to cellular communications, from secure Web access to digital cash, cryptography is an essential part of today's information systems. Cryptography helps provide accountability, fairness, accuracy, and confidentiality. It can prevent fraud in electronic commerce and assure the validity of financial transactions. It can prove your identity or protect your anonymity. It can keep vandals from altering your Web page and prevent industrial competitors from reading your confidential documents.
Electronic commerce schemes will face fraud, through forgery, misrepresentation, denial of service, and cheating. In fact, computerization makes the risks even greater, by allowing attacks that are impossible against non-automated systems. Privacy violations are another threat. Some attacks on privacy are targeted: a member of the press tries to read a public figure’s e-mail, or a company tries to intercept a competitor’s communications. Others are broad data-harvesting attacks, searching a sea of data for interesting information: a list of rich widows, AZT users, or people who view a particular Web page.

Criminal attacks are often opportunistic, and often all a system has to be more secure than the next system. But there are other threats. Some attackers are motivated by publicity; they usually have significant resources via their research institution or corporation and large amounts of time, but few financial resources.

The good news about cryptography is that we already have the algorithms and protocols we need to secure our systems. The bad news is that that was the easy part; implementing the protocols successfully requires considerable expertise. The areas of security that interact with people—key management, human/computer interface security, access control—often defy analysis. And the disciplines of public-key infrastructure, software security, computer security, network security, and tamper-resistant hardware design are very poorly understood.

The use of cryptography is fundamental to providing the security required for Internet commerce. Cryptography encompasses encrypting data for privacy, providing reliable means of verifying identity, recording a digital signature, and ensuring that messages and other documents have not been tampered with.

In the future, as commerce and communications continue to move to computer networks, cryptography will become more and more vital

1.2 AUTHENTICATION

Authentication – reliable verification of identity – is frequently required by both the merchant and the customer at various stages in electronic commerce. It is the process of establishing identity as an individual, a function, or a member of a class of individuals. The first example of this requirement arises at the initial point of contact between the merchant and the customer, the electronic shop-front. The customers entering this electronic shopping environment may need confirmation, that the customer is genuinely in contact with the company customer has selected and not an electronic impostor. Authentication of the customer to the merchant is dependent on the specifics of each customer merchant relationship.

Suppose a consumer wants to use a credit card to buy something over the Internet. The merchant should be able to determine if the credit card is valid. Digital certificates (a structure that contains a public value or a key that is bound to an identity – merchant or customer) can be used to verify that a consumer is authorized to use the offered credit card in an Internet based transaction. A consumer obtains a digital certificate from a financial institution that issues the credit card. The digital certificate contains the identity of the consumer and the consumer’s public key.
1.3 DATA INTEGRITY

A risk of any electronic communication is that the data sent from one party to another may be corrupted in transit. In electronic commerce applications, a corruption in information can have unintended financial consequences. Mathematical algorithms that employ message digest and digital signature can be used to ensure the integrity of data in transit against random corruption and malicious tampering. The digital signature can be verified by the merchant to provide both authentication and non-repudiation. The merchant also computes a message digest of the order and compares it with the received message digest to assure data integrity.

Authentication and Data Integrity are dealt in more details in a separate chapter. Confidentiality is achieved via encryption and cryptography, the methods of which are dealt in chapters on Symmetric Key Cryptography and Public Key Cryptography. Some systems discussed for Public Key Cryptography also provide for authentication and data integrity along with the secure means of key exchange.

There are a myriad of reasons to take security seriously, not just for the secure transactions without any frauds but also for ensuring privacy of the individual customer venturing on the Internet.
2.0 ELECTRONIC TRANSACTIONS AND SECURITY

The purpose of this paper is to survey the security methods and products for data transport, especially as it relates to E-commerce. Before we do that it is perhaps essential to understand the basics of secure electronic transactions and explanation of the terms frequently used.

2.1 BASIC TERMINOLOGY

Suppose that someone wants to send a message to a receiver, and wants to be sure that no-one else can read the message. However, there is the possibility that someone else might open the letter or hear the electronic communication. In cryptographic terminology, the message is called plaintext or cleartext. Encoding the contents of the message in such a way that hides its contents from outsiders is called encryption. Encryption is one of the major methods of ensuring data security. Encryption can also be defined as a transformation of information in any form (text, video, graphics etc.) into another representation using complex function (algorithm) and a special encryption key (a string). The encrypted message is called the ciphertext. The process of retrieving the plaintext from the ciphertext is called decryption. Encryption and decryption usually make use of a key, and the coding method is such that decryption can be performed only by knowing the proper key.

Cryptography is the art or science of keeping messages secret. Cryptanalysis is the art of breaking ciphers, that is, retrieving the plaintext without the proper key. Cryptography deals with all aspects of secure messaging, authentication, digital signatures, electronic money, and other applications. Cryptology is the branch of mathematics that studies the mathematical foundations of cryptographic methods.

2.2 EXAMPLE OF A SECURE ELECTRONIC TRANSACTION

A familiar process of writing and sending a check can be used to illustrate the transaction process and to better understand how cryptography is used to secure electronic communications.

2.2.1 THE TRADITIONAL CHECK WRITING PROCESS

Let’s look at the steps involved in a familiar banking transaction to ensure that we have a secure exchange of information when sending a check.

- You start with a blank check and fill in the receiver’s name and the amounts.
- You then sign the check to authorize the transfer of funds.
- You place it in an envelope to secure it from undesired viewing and pass it on to the postal system for delivery.
- When the check is received the envelope may be checked for tampering to ensure the integrity of the data.
- If there is any doubt about authenticity, the signature can be verified against a signature card on file at the bank.
- If the signature matches, the bank has established a legally binding transaction.
2.2.2 THE ELECTRONIC VERSION

The simplest electronic version of the check can be text, created with a word processor or e-mail package, asking your bank to pay someone a specific sum. However sending this “check” over an electronic network creates several security problems:

- Since anyone could intercept and read the file we need privacy.
- Since someone else could create a similar counterfeit file we need authentication.
- Since the originator could deny creating the file we need non-repudiation.
- Since someone could alter the file we need integrity.

Following steps need to be taken to properly execute the Electronic version of our check writing.

Privacy and Encryption

One of the key aspects is the privacy. This can be achieved via encryption. The electronic check can be encrypted using a high-speed mathematical transformation with a key that will be used later to decrypt the document. This is often referred to as a “symmetric” key system because the same key is used for encryption as well as the decryption process. Since the encrypted document needs a key which is extremely difficult to guess randomly, the document can now be considered to be safe.

As the check is sent over the network it is unreadable without the key. But the problem is the receive of this document will not be able to decrypt the contents without the key itself. This key now has to be delivered to the receiver over the same network. The next challenge is to deliver the symmetric key in a secure fashion.

So we are still stuck with the problem of safely delivering the secret key. This has been completely addressed by something called “Public-Key Cryptography”.

Public Key Cryptography for Delivering Symmetric Keys

The introduction of public-key encryption in the late 70’s solved the problem of delivering the symmetric encryption key to the target destination in a secure manner. With public-key cryptography there are two keys: one is kept private and the mate is made publicly available. What is encrypted with one can only be decrypted with the other. The thing to be kept in mind is that revealing the “public” key does not in any way compromise the “private” key. No other private key can decrypt the data that is encrypted with your public key.

Unfortunately public-key algorithms are relatively very slow but they are ideal for encrypting small amounts of information such as a symmetric key! So the check is encrypted with a fast symmetric key (uniquely generated for this occasion). Then the symmetric key is encrypted with the receiver’s public key. Now only the private key of the receiver can recover the symmetric key and thus decrypt the check.
This solves one major problem of security since now we have a check that is encrypted and can be read only by the target recipient. We have just created a digital version of the envelope. However there is still one problem. Unfortunately, someone else could have created the same document using the public key of the recipient and e-mailed a bogus check. To prevent this we need the digital version of a signature to make sure that the encrypted document has been sent by the right person. This problem of verification of the source is termed as authentication.

Writing and sending a check using Public-Key Cryptography

Digital Signature

The same public and private-key system used for the digital envelope can be used to guarantee the identity of the originator. We simply switch the roles of the public and private keys. Anything encrypted with your private key can be decrypted by anyone else using your public key. It must have come from you because only you have your private key.

The process of digitally signing starts by taking a mathematical summary (called a “hash”) of the check. The hash code is like a “fingerprint” of the check. If even a single bit of the check changes, the hash code will dramatically change. The next step in creating a digital signature is to encrypt the hash code with your private key. Finally the encrypted fingerprint is appended to the check.

In order to verify that the check came from you the receiver of your check can decrypt the hash code sent by you using your public key. At the same time the hash code can be recreated from the received check and compared with the original hash code. If the hash codes match, then the receiver has verified that the check has not been altered. The receiver also knows that only you could have sent the check because no one else has the private key that encrypted the original hash code. Thus
there are two aspects to the verification – verification of the decrypted hash code and the ability to decrypt the hash code itself using the public key.

Now we have a system that can encrypt your check, secure the keys and authenticate the sender.

The last piece of the puzzle is to guarantee that the public-key list (used by everyone to verify signatures) has not been tampered with. This would be equivalent to warranting that the signature card you have on file at the bank has not been switched with a forgery.

This is done by having the public directory entries digitally “signed” by the system administrator. Using the system administrator’s public key anyone can verify that the directory entry is genuine. The signed directory entry is known as a “certificate.”

2.4 IMPORTANCE OF PUBLIC KEY TECHNOLOGY

The process of encryption involves taking the “plain text” source information and using a mathematical algorithm to “scramble” the information into a form where it looks incomprehensible. As explained above the mathematical process is controlled by a certain number or character string called a “key”. An authorized recipient who has knowledge of both the algorithm used and the key, is able to use these to reverse the scrambling process and recover the original source information. Knowledge of either the algorithm or the key alone is not sufficient to enable decryption.

Conventional key encryption systems suffer from several shortcomings that make their use in large distributed cyberspace communities cumbersome and impractical. All of these problems center around the issue of creating, communicating and protecting the keys.

Most encryption technologies use what is called a “symmetric” key. That means that when one key is used to perform the encryption operation, the same key is used to reverse it. More on this is explained later in this paper. This leads to an immediate weakness in security. Both the originator and recipient must know the same key, either by pre-arrangement or by communicating the key in parallel with the encrypted message.

2.4.1 KEY PROLIFERATION

In a symmetric key system, each and every discrete originator/recipient pair must have a unique key for their communication. The number of keys required grows exponentially with the number of users. (For n users, the number of unique keys required is n^2-n). This is not a problem for small numbers of users, but it quickly becomes unmanageable in a large enterprise environment (an enterprise environment is defined as a situation when a large number of users are communicating with each other as is often the case in corporate environment and scientific communities). There is the problem of requiring each user to keep track of a large number of keys independently and knowing which key applies to which files. There is the problem of updating keys – security policy or a suspected breach may require that specific user’s keys must be changed. The difficulties involved in notifying a large number of users about key changes, simultaneously but securely, are not trivial. Any form of centralized directory of users/keys is itself a threat to security.
2.4.2 FILE PROLIFERATION

In any enterprise environment we can expect to find groups of users who must share access to the same body of information. This might be working groups, multiple levels of management or individuals who have similar functions at different locations. In order to ensure the privacy of information, but at the same time make it available to multiple recipients, an originator must encrypt a copy of the information uniquely for each intended recipient. For any group of more than a few users, this quickly leads to some insurmountable problems. There is the proliferation of multiple copies of the same file and the attendant cost of storage. There are concurrency (version control) issues. If the original is changed the changes must be propagated reliably to every recipient, so every copy must be repeated and every user must be able to know which is the most current version. There is also the problem of keeping track of which encrypted copy belongs to which user.

In short, the problems that have prevented encryption technologies from being widely employed by enterprises are based not on the encryption algorithms themselves (many are reasonably secure for commercial purposes) but on the difficulty of managing large numbers of keys.

2.4.3 SOLUTION: PUBLIC KEY

Public keys solve two of the four security problems mentioned at the beginning of this paper which are confidentiality and authentication.

- When you encrypt a message with a public key, which is known to everyone, we know that it can only be decrypted by that one person who holds the decryption key. Therefore, this confirms confidentiality and privacy, since no one can read it other that the person you sent the message to. This solves the problem of key proliferation since now we need only 2n keys instead of $n^2 - n$ keys.
- When a person sends you a message that was encrypted with a private key, this assures you of their identity (authentication), because you will only be able to read the message, by decrypting it with the public key of that person.

Public Key Encryption technology addresses the issues of key proliferation and key protection. The enabling factor in public key systems is called Asymmetric Encryption. Each operation requires a unique pair of keys. One key is used to perform the encryption operation. However, unlike symmetric systems, that same key cannot be used to decrypt. Only the other key of the matched pair will work for decrypting the information.

In a public key system, each user requires only two keys: a Public Key and a Private Key. Public keys have no security requirement and can be published or distributed at will. When a user wishes to secure information for another user, the sender acquires the recipient’s public key and uses that to encrypt the content. The recipient then uses his own private key, which is known only to himself, to decrypt.
Any number of users can provide information securely for one specific recipient by using that person’s public key. Only the intended recipient can access the information by using the matching private key. Thus each user in a public key system owns only two keys, regardless of the number of correspondents involved. The key proliferation problem is mitigated. The number of keys needed for n users is now 2n (as compared to n^2-n for symmetric systems). The key protection issue is simplified. Public keys need not be protected because they cannot be used to decrypt anything. Hence, they can be published to all users. Private keys are known only to their owners and need not be communicated to any other user for any reason. This eliminates the security risk associated with passing keys along with the encrypted messages.

Public key technology also addresses the file proliferation issue in a similar fashion. The file to be distributed can be encrypted using a secret symmetric key and placed in a location for others to get it. The secret key can then be sent securely to all the group members (with whom this file is to be shared) using their public keys. This way only one file has to be maintained.

### 2.4.4 KEY MANAGEMENT

Solid Public Key encryption can be had from many sources. It is the ability to manage the key pairs which really tests the software package.

Simply limiting the proliferation of keys is not by itself a complete solution. There are still many difficult issues involved in handling the keys themselves:

- Who is permitted to create key pairs within an organization?
- How are they stored and communicated?
- How do we prove authenticity?
- How do we recover data encrypted with keys that are lost or destroyed?
- Like a stolen credit card, how do we tell the world that a key has been compromised and should no longer be accepted?
- How do we change or cancel a key pair when an employee leaves yet still be able to decrypt their data later?
• How do we communicate key pair changes?
• How do we publish your Public Key for others to use with assurance that it really is your key?
• If a widely available database/yellow pages of Public Keys is used, who controls the updates?

Philip Zimmermann, author of PGP, said in his Users Guide: “This whole business of protecting public keys from tampering is the single most difficult problem in practical public key application. It is the Achilles’ heel of public key cryptography and a lot of software complexity is tied up in solving this one problem”

There are many different types of software on the market specifically geared for encryption and decryption. Most of them use some standard methods, most of which are described below. To recapitulate, there are two classes of key-based algorithms, symmetric (or secret-key) and asymmetric (or public-key) algorithms. The difference is that symmetric algorithms use the same key for encryption and decryption (or the decryption key is easily derived from the encryption key), whereas asymmetric algorithms use a different key for encryption and decryption, and the decryption key cannot be derived from the encryption key. In the next two chapters specific methods for each of these categories are discussed.
3.0 SYMMETRIC KEY ALGORITHMS

3.1 SYMMETRIC OR SECRET KEY

\[
\text{Plaintext} \xrightarrow{\text{Encryption}} \text{Ciphertext} \xrightarrow{\text{Decryption}} \text{Plaintext}
\]

Symmetric key method

Symmetric key algorithms are designed to be fast and are much easier to use than other methods. However, this system has a problem of key distribution, because the same key is needed to encrypt and decrypt the message. Symmetric algorithms can be divided into stream ciphers and block ciphers. Stream ciphers can encrypt a single bit of plaintext at a time, whereas block ciphers take a number of bits (typically 64 bits), and encrypt them as a single unit. Following is a brief survey of the various symmetric key algorithms that are currently being used or have been proposed.

3.2 ALGORITHMS

3.2.1 DES: DATA ENCRYPTION STANDARD

DES, an acronym for the Data Encryption Standard, is the name of the Federal Information Processing Standard (FIPS) 46-1, which describes the data encryption algorithm (DEA). The DEA is also defined in the ANSI standard X9.32. The DEA, often called DES, has been extensively studied since its publication and is the best known and widely used symmetric algorithm in the world. It was made a standard by the US government.

The DES is a block algorithm with a 64-bit block size and uses a 56-bit key during execution (8 parity bits are stripped off from the full 64-bit key). The DES is a symmetric cryptosystem, 16-round Feistel cipher and was originally designed for implementation in hardware. When used for communication, both sender and receiver must know the same secret key, which can be used to encrypt and decrypt the message, or to generate and verify a message authentication code (MAC). The DES can also be used for single-user encryption, such as to store files on a hard disk in encrypted form. In a multi-user environment, secure key distribution may be difficult; public-key cryptography provides an ideal solution to this problem. DES is regarded by many to be a strong algorithm.

The development of AES, the Advanced Encryption Standard is underway. AES will replace DES.

3.2.2 TRIPLE-DES

This is the same as DES, except that it was made to provide better security than DES. This is done by encrypting the message three times with three different keys.
A number of modes of triple-encryption have been proposed:

**DES-EEE3**: Three DES encryptions with three different keys.
**DES-EDE3**: Three DES operations in the sequence encrypt-decrypt-encrypt with three different keys. **DES-EEE2** and **DES-EDE2**: Same as the previous formats except that the first and third operations use the same key.

<table>
<thead>
<tr>
<th># of Encryptions</th>
<th># of Keys</th>
<th>Computations</th>
<th>Storage</th>
<th>Type of attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>single</td>
<td>1</td>
<td>256</td>
<td>-</td>
<td>known plaintext</td>
</tr>
<tr>
<td>single</td>
<td>1</td>
<td>238</td>
<td>238</td>
<td>chosen plaintext</td>
</tr>
<tr>
<td>single</td>
<td>1</td>
<td>-</td>
<td>256</td>
<td>chosen plaintext</td>
</tr>
<tr>
<td>double</td>
<td>2</td>
<td>2112</td>
<td>-</td>
<td>known plaintext</td>
</tr>
<tr>
<td>double</td>
<td>2</td>
<td>256</td>
<td>256</td>
<td>chosen plaintext</td>
</tr>
<tr>
<td>double</td>
<td>2</td>
<td>-</td>
<td>2112</td>
<td>chosen plaintext</td>
</tr>
<tr>
<td>triple</td>
<td>2</td>
<td>256</td>
<td>256</td>
<td>known plaintext</td>
</tr>
<tr>
<td>triple</td>
<td>2</td>
<td>2120-t</td>
<td>2t</td>
<td>known plaintext</td>
</tr>
<tr>
<td>triple</td>
<td>2</td>
<td>-</td>
<td>256</td>
<td>chosen plaintext</td>
</tr>
<tr>
<td>triple</td>
<td>3</td>
<td>2112</td>
<td>256</td>
<td>known plaintext</td>
</tr>
<tr>
<td>triple</td>
<td>3</td>
<td>256</td>
<td>2112</td>
<td>chosen plaintext</td>
</tr>
</tbody>
</table>

**Comparison of different forms of DES multiple encryption**

Both the DES and the triple-DES are currently being used a lot by financial industries. The most secure form of multiple encryption is triple-DES with three distinct keys.

### 3.2.3 IDEA

IDEA – International Data Encryption Algorithm – uses a 128-bit key and is believed to be quite strong, and is used by a popular program PGP to encrypt files and electronic mail. IDEA (International Data Encryption Algorithm) is the second version of a block cipher designed and presented by Lai and Massey. It is a 64-bit iterative block cipher with a 128-bit key. The encryption process requires eight complex rounds. While the cipher does not have a Feistel structure, decryption is carried out in the same manner as encryption once the decryption subkeys have been calculated from the encryption subkeys. The cipher structure was designed to be easily implemented in both software and hardware, and the security of IDEA relies on the use of three incompatible types of arithmetic operations on 16-bit words. However some of the arithmetic operations used in IDEA are not that fast in software. As a result the speed of IDEA in software is similar to that of DES.

One of the principles used during the design of IDEA was to facilitate analysis of its strength against differential cryptanalysis and IDEA is considered to be immune to differential cryptanalysis. Furthermore there are no linear cryptanalytic attacks on IDEA and there are no known algebraic weaknesses in IDEA. IDEA is generally considered to be a very secure cipher and both the cipher development and its theoretical basis have been openly and widely discussed.
3.2.4 AES

The AES is the Advanced Encryption Standard and is intended to replace DES. While reports over the last few years of the demise of DES have been greatly exaggerated, most now agree this venerable cipher is approaching the end of its useful life. DES will not be reaffirmed as a federal standard after 1998. On January 2, 1997 the AES initiative was announced and on September 12, 1997 the public was invited to propose suitable block ciphers as candidates for the AES. NIST is looking for a cipher that will remain secure well into the next century. For more information see the home page for AES at http://csrc.nist.gov/encryption/aes/aes_home.htm.

There is considerable interest in the AES initiative and 15 candidates were accepted for consideration in the first round. Among these were close variants of some of the more popular and trusted algorithms currently available, such as CAST, RC5 and SAFER-SK. Other good candidates from well-respected cryptographers were also submitted. The reason for close variants being proposed rather than the original ciphers is that one of the criteria for the AES submission is the ability to support 128-bit blocks of plaintext. Most ciphers were developed with an eye to providing a drop-in replacement for DES and, as a result, were often limited to having a 64-bit block size.

A recently published status report by NIST on the development of Advanced Encryption Standard shows the completion of first round of analysis of the 15 candidates. NIST has reviewed the results of this research and selected the following five algorithms as finalists.

MARS – developed by International Business Machines Corp. of Armonk, NY
RC6™ – developed by RSA Laboratories of Bedford, Mass.
Rijndael – developed by Joan Daemen and Vincent Rijmen of Belgium
Serpent – developed by Ross Anderson, Eli Biham and Lars Knudsen of the UK, Israel and Norway resp.
Twofish – developed by Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall and Niels Ferguson.

Other algorithms considered were: DEAL, FROG, HPC (Hasty Pudding Cipher), LOKI97, MAGENTA, DFC (Decorrelated Fast Cipher), E2, CAST-256, SAFER®, and CRYPTON. It is expected that by the end of summer 2001 NIST will declare an encryption standard.

3.2.5 RC2, RC4, AND RC5

These were all developed by Ronald Rivest for RSA Data Security. RC2 is a variable key-size block cipher. It is faster than DES and is designed as a “drop-in” replacement for DES. RC4 is a variable key-size stream cipher with byte-oriented operations. The algorithm is based on the use of a random permutation. Eight to sixteen machine operations are required per output byte. RC5 is a parameterized algorithm with a variable block size, a variable key size, and a variable number of rounds. Typical choices for the block size are 32 bits (for experimentation and evaluation purposes only), 64 bits (for use a drop-in replacement for DES) or 128 bits. The number of rounds can range from 0 to 255. The key can range from 0 bits to 2048 bits in size. Such built-in variability provides flexibility at all levels of security and efficiency. The RC2 and RC4 key length is limited to 40 bits.
3.2.6 SAFER

SAFER is an algorithm developed by J.L. Massey (one of the developers of IDEA). It is claimed to provide secure encryption with fast software implementation even on 8-bit processors. Two variants are available, one for 64 bit keys and the other for 128 bit keys. SAFER+ is an algorithm proposed as an AES to replace the current DES standard. SAFER+ has a good security margin and supports on-the-fly subkey generation with subkeys computable in any order. But it is a slow algorithm and has been found to be vulnerable to certain types of attacks.

3.2.7 CRYPTOGRAPHIC HASH FUNCTIONS

MD5 (Message Digest Algorithm 5) is a secure has algorithm developed at RSA Data Security, Inc. The MD5 hashing algorithm is designed so that if the message is changed in any way, the message digest will also change (and it would be extremely difficult to find a way to change the message while still producing the same message digest). It can be used to hash an arbitrary length byte string into a 128 bit value. MD5 is in wide used, and is considered reasonably secure.

MD2 and MD4 are also message-digest algorithms developed by Rivest. They, like MD5, are meant for digital signature applications where a large message has to be “compressed” in a secure manner before being signed with the private key. All three algorithms take a message of arbitrary length and produce a 128-bit message digest. While the structures of these algorithms are somewhat similar, the design of MD2 is quite different from that of MD4 and MD5. MD2 was optimized for 8-bit machines, whereas MD4 and MD5 were aimed at 32-bit machines.

SHA (Secure Hash Algorithm) is a cryptographic hash algorithm published by the United States Government. It produces an 160 bit hash value from an arbitrary length string. The algorithm takes a message of less than 264 bits in length and produces a 160-bit message digest. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks. This algorithm though fairly new is considered to be good.

Tiger and RIPEMD-160 are two other new hash algorithms.

A number of symmetric algorithms are available from ftp.funet.fi:/pub/crypt/cryptography/symmetric.
Public key algorithms use a different key for encryption and decryption, and the decryption key cannot be derived from the encryption key. Public key methods are important because they can be used to transmit encryption keys or other data securely even when the parties have no opportunity to agree on a secret key in private. All known methods are quite slow, and they are generally only used to encrypt session keys (randomly generated “normal” or “symmetric” keys), that are then used to encrypt the bulk of the data using a symmetric cipher.

This section covers the most commonly used public key cryptographic methods and systems.

4.1 DIFFIE-HELLMAN KEY EXCHANGE

This is not a method for encrypting and decrypting, but a way to develop and exchange a shared private key over a public network. The Diffie-Hellman key agreement protocol (also called exponential key agreement) was developed by Diffie and Hellman (*ref) in 1976 and published in the groundbreaking paper “New Directions in Cryptography.” It is generally considered to be secure when sufficiently long keys and proper generators are used. The security of Diffie-Hellman relies on the difficulty of the discrete logarithm problem – a problem equivalent to factoring large integers.

The protocol has two system parameters $p$ and $g$. They are both public and may be used by all the users in a system. Parameter $p$ is a prime number and parameter $g$ (usually called a generator) is an integer less than $p$, with the following property: for every number $n$ between 1 and $p-1$ inclusive, there is a power $k$ of $g$ such that $g^k = n \mod p$. The protocol depends on the discrete logarithm problem for its security. Diffie-Hellman is sensitive to the choice of the strong prime and the generator. The size of the secret exponent is also important for its security.

The Diffie-Hellman key exchange is vulnerable to a man-in-the-middle attack. In this attack, an opponent Carol intercepts Alice’s public value and sends her own public value to Bob. When Bob transmits his public value, Carol substitutes it with her own and sends it to Alice. Carol and Alice thus agree on one shared key and Carol and Bob agree on another shared key. After this exchange, Carol simply decrypts any messages sent out by Alice or Bob, and then reads and possibly modifies them before re-encrypting with the appropriate key and transmitting them to the other party. This vulnerability is present because Diffie-Hellman key exchange does not authenticate the participants. Possible solutions include the use of digital signatures and other protocol variants such as Station-to-Station (STS) protocol which is an authenticated Diffie-Hellman key agreement protocol.

Recent analysis of the original Diffie-Hellman protocol has led to an understanding that it is an example of a much more general cryptographic technique, the common element being the derivation of a shared secret value (a key) from one party’s key and another party’s private key. The parties’ key pairs may be generated anew at each run of the protocol, as in the original Diffie-Hellman protocol. The public keys may be certified, so that the parties can be authenticated and there may be a combination of these attributes.
4.2 RSA: RIVEST, SHAMIR AND ADLEMAN

This is a data encryption scheme that can provide both encryption and authentication. This is the most commonly used public key algorithm. The key may be any length depending on how you use it. It is generally considered to be secure when sufficiently long keys are used (768 or 1024 bit keys). The RSA algorithm is the best known of the ‘integer factorization’ family of ciphers where the strength of the cipher lies in the mathematical difficulty in factoring large integers. The security of RSA relies on the difficulty of factoring large integers. In this scheme integers of large enough size are chosen so as to make ‘brute-force’ factorization infeasible in the absence of any faster or better mathematical or computational techniques. Dramatic advances in the method of factoring would make RSA more vulnerable.

For now integers of some 768 binary bits are seen as generally filling this requirement although an increase to 1024 bits is now accepted as necessary given the advances in computation techniques and computer speeds. For more advanced work and for government or high-risk usage, integers of 2048 bits are being requested.

For RSA security offered is considered to be dependent upon mathematical difficulty inherent in factoring large numbers. RSA is currently the most important public key algorithm. RSA can also be used as the basis of a digital signature. The RSA scheme can be characterized as having the following properties:

<table>
<thead>
<tr>
<th>Scheme</th>
<th>RSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘Hard Problem’</td>
<td>Difficulty in factoring large integers</td>
</tr>
<tr>
<td>Key Generation</td>
<td>Select two large prime numbers p and q that define the modulus n, where n=p<em>q. Select integers e and d, where e</em>d = 1 [Mod (p-q)*(q-1)]</td>
</tr>
<tr>
<td>Public Key</td>
<td>Two large integers – encryption key e, and modulus n</td>
</tr>
<tr>
<td>Private Key</td>
<td>Two large integers – decryption key d, and modulus n.</td>
</tr>
<tr>
<td>Services</td>
<td>Confidentiality</td>
</tr>
<tr>
<td></td>
<td>Encryption: use public key</td>
</tr>
<tr>
<td></td>
<td>Decryption: use private key</td>
</tr>
<tr>
<td></td>
<td>Authenticity</td>
</tr>
<tr>
<td></td>
<td>Create signature: hash + exponentiation using private key.</td>
</tr>
<tr>
<td></td>
<td>Verify signature: hash + exponentiation using public key</td>
</tr>
</tbody>
</table>

4.3 ELGAMAL (DISCRETE LOGARITHM SYSTEMS)

ElGamal first proposed a cipher system based on the so-called ‘discrete logarithm’ problem. The operations in this scheme are performed over mathematical finite groups. The discrete log problem over a finite group G is defined as: Given g and y in G such that \( y = g^x \), find x. If the number of elements in G that can be written in the form \( g^x \) is sufficiently large, the solution of the discrete logarithm problem is intractable. The ElGamal scheme is an example of a discrete logarithm system and works for non-zero integers modulo a large prime number.
Analysis based on the best available algorithms for both factoring and discrete logarithms show that RSA and ElGamal have similar security for equivalent key lengths. The main disadvantage of ElGamal is the need for randomness, and its slower speed (especially for signing). Another potential disadvantage of the ElGamal system is that message expansion by a factor of two takes place during encryption. However, such message expansion is generally unimportant if the cryptosystem is used only for exchange of secret keys.

Discrete logarithms or ElGamal systems can also be used for encryption or digital signatures in a way similar to the RSA. One of the most widely used schemes is the USA’s Digital Signature Algorithm (DSA) – a part of its Digital Signature Standard (DSS). Its design has not been made public, and there seem to be some potential problems with it (e.g., leaking hidden data in the signature, and revealing your secret key if you ever happen to sign two different messages using the same random number.) ElGamal DSS scheme can be summarized as follows:

### Scheme: ElGamal

**‘Hard Problem’:** Difficulty in solving the discrete logarithm problem

**Key Generation:** Select a large prime number $p$ with generator $g$ of the group of non-zero integers modulo $p$. Choose a secret number $a$ as the private key and calculate public key $y$, $y = g^a \text{ Mod}(p)$. A distinct random secret number $k$, $I = k \times p - 1$ is chosen as part of the key for each encryption.

**Public Key:** Group elements – $g$, $y$ and modulus $p$. DSS also requires another prime integer $q$ that divides $p-I$, as part of its public key.

**Private Key:** Secret number $a$.

**Services:** Confidentiality (using ElGamal)

- Encryption: use public key and secret random number $k$
- Decryption: use private key

Authenticity (using DSS)

- Create signature: use secret random number $k$ and public key including additional parameter $q$.
- Verify Signature: find a related message to the signature and verify relationship using public key

### 4.4 ELLIPTIC CURVE ALGORITHMS

Elliptic curve cryptosystems were first proposed independently by Victor Miler and Neal Koblitz in the mid-1980s. At a high level, they are analogs of existing public-key systems in which modular arithmetic is replaced by operations defined over elliptic curves. The elliptic curve cryptosystems that have appeared in the literature can be classified into two categories according to whether they are analogs to RSA or discrete logarithm (ElGamal) based systems.

Just as in all public-key cryptosystems, the security of elliptic curve cryptosystems relies on the underlying hard mathematical problems. It turns out that elliptic curve analogs of RSA are mainly of academic interest and offer no practical advantage over ordinary RSA, since their security is
based on the same underlying problem as RSA, namely integer factorization. But at this point elliptic curve based RSA system is still considered to offer better security. The situation is quite different with elliptic curve variants of discrete logarithm based systems. The security of such systems depends on the following hard problem: Given two points \( G \) and \( Y \) on an elliptic curve such that \( Y = kG \) (i.e., \( Y \) is \( G \) added to itself \( k \) times), find the integer \( k \). This problem is commonly referred to as the “elliptic curve discrete logarithm problem.”

Presently, the methods for computing general elliptic curve discrete logs are much less efficient than those for factoring or computing conventional discrete logs. As a result, shorter key sizes can be used to achieve the same security of conventional public-key cryptosystems, which might lead to better memory requirements and improved performance. One can easily construct elliptic curve encryption, signature, and key agreement schemes by making analogs of ElGamal, RSA, and Diffie-Hellman. These variants appear to offer certain implementation advantages over the original schemes, and they have recently drawn more and more attention from both the academic community and the industry.

The main attraction of elliptic curve cryptosystems over other public-key cryptosystems is the fact that they are based on a different, hard problem. This could lead to smaller key sizes and better performance in certain public-key operations for the same level of security. Roughly speaking elliptic cryptosystems with a 160-bit key offer the same security of RSA and discrete logarithm based systems with a 1024-bit key. As a result, the length of the public key and private key is much shorter in elliptic curve cryptosystems. Moreover elliptic curve systems are considered to be faster than the corresponding discrete logarithm based systems. Elliptic curve systems are faster than RSA in signing and decryption, but slower than RSA in signature verification and encryption.

With academic advances in attacking different hard mathematical problems both the security estimates for various key sizes in different systems and the performance comparisons between systems are likely to change.

### 4.5 COMPARISONS OF ENCRYPTION SYSTEMS

**Comparison of some of the encryption systems available on the Internet**

<table>
<thead>
<tr>
<th>System</th>
<th>Algorithms/Methods</th>
<th>Provides</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGP</td>
<td>IDEA, RSA, MD5</td>
<td>confidentiality, authentication, integrity &amp; non-repudiation</td>
</tr>
<tr>
<td>S/MIME</td>
<td>User- specified</td>
<td>confidentiality, authentication, integrity &amp; non-repudiation</td>
</tr>
<tr>
<td>SSL</td>
<td>RSA, RCZ, RC4, MD5 &amp; others</td>
<td>confidentiality, authentication, integrity &amp; non-repudiation</td>
</tr>
<tr>
<td>PCT</td>
<td>RSA, MD5, RCZ, RC4 &amp; others</td>
<td>confidentiality, authentication, integrity &amp; non-repudiation</td>
</tr>
<tr>
<td>Protocol</td>
<td>Key/Algorithm</td>
<td>Features</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>S-HTTP</td>
<td>RSA, MD5 &amp; others</td>
<td>Confidentiality, authentication, integrity &amp; non-repudiation; however, it's obsolete</td>
</tr>
<tr>
<td>SET &amp; CyberCash</td>
<td>RSA, MD5, RC2</td>
<td>Confidentiality, integrity, authentication, nonrepudiation</td>
</tr>
<tr>
<td>DNSSEC</td>
<td>RSA, MD5</td>
<td>Authentication, integrity</td>
</tr>
<tr>
<td>Ipsec &amp; IPV6</td>
<td>Diffie-Hellman &amp; others (optional), authentication, integrity</td>
<td></td>
</tr>
<tr>
<td>Kerberos</td>
<td>DES</td>
<td>Confidentiality, authentication</td>
</tr>
<tr>
<td>SSH</td>
<td>RSA, DES, triple-DES, Diffie-Hellman, Blowfish &amp; others</td>
<td>Confidentiality, authentication</td>
</tr>
</tbody>
</table>

Ref: (Table from Web Security & Commerce, by O'Reilly & Associates, Inc. pgs 218-19)
5.0 PRODUCTS, ANALYSIS, AND COMPARISONS

Security of Electronic Commerce seems to be a booming business as there is wide array of products available addressing various aspects of E-commerce. Obviously not all products could be reviewed and the long list had to be truncated quite a bit. Products of only a few companies have been included here. In the following sections these products are listed under the names of their companies. Given below are the companies whose security products have been reviewed.

1. Cybersafe
2. iD2
3. Baltimore
4. Netscape
5. Entrust
6. RSA
7. VeriSign
8. Entegrity Solutions
9. Xcert
10. Certicom
11. IBM
12. Kerberos

5.1 CYBERSAFE

5.1.1 TRUSTBROKER™ SECURITY SERVER

TrustBroker™ Security Suite helps large, distributed corporate networks address the security and accessibility issues of global networked organizations. This powerful software product allows organizations to leverage existing infrastructure investments and deploy security solutions today that are maintainable and extensible into the future. TrustBroker Security Suite runs on a variety of platforms; supports both Public Key and Kerberos authentication mechanisms; enables secure single sign-on; and is adapted to take advantage of devices such as token cards and smart cards.

TrustBroker Security Server acts as the authentication engine for TrustBroker Security Suite issuing secure credentials that are brokered to enable secure single sign-on or login. TrustBroker Security Server creates an interoperability mechanism between secret key and Public Key infrastructures by allowing authentication via both technologies. TrustBroker Security Server provides the foundation for many security solutions. Other products in the TrustBroker Security Suite work in conjunction with TrustBroker Security Server to complete these solutions.

- TrustBroker Client for Windows or TrustBroker Client for UNIX – The client component of TrustBroker Security Suite allows the user to authenticate to TrustBroker Security Server and use retrieved credentials to enable secure single sign-on to a multitude of applications and network services.
- TrustBroker MVS – Accepts TrustBroker credentials and utilizes existing MVS authorization systems to securely generate one-time use, limited-life password replacements known as passtickets.
• TrustBroker Developer Pack – Provides software development kits (SDKs) that utilize credentials retrieved from TrustBroker Security Server to enable secure single sign-on to internally developed mission critical applications.

The TrustBroker Security Server architecture provides a foundation upon which an organization’s security architecture can be built.

Features of TrustBroker Security Server:
• Supports both Public Key and secret key protocols
• Supports multiple levels of encryption
• Secret Key authentication: Kerberos v5.4
• PKI – Authentication with physical smart card
• Supports multiple PK certificate authorities
• Grants secret key credentials based on PK authentication
• Supports Authentications by Entrust, Verisign, Microsoft, Netscape
• Data Encryption Algorithms: DES, Triple-DES
• Hash Algorithms: SHA-1, MD5, CRC
• Certificates: X.509 v3
• Interfaces: LDAP, CMS
• OS: UNIX, Windows NT

Application: E-mail, SAP, Lotus Notes™, Oracle, Sybase, telnet, rsh, rcp, rlogin, ftp.

Summary: TrustBroker Security Server creates an interoperability mechanism between secret key and Public Key infrastructures by allowing authentication via both technologies making it unnecessary to choose one security solution over another. Users can authenticate to and use both secret key enabled network services and Public Key enabled services such as secure Web sites and e-mail.

5.1.2 DEFENSOR® PRODUCT SUITE

Classification: PKI/ Data Transport

Defensor® products allow secure communications between clients, servers, gateways, and mainframes within an enterprise regardless of the network technology or the geographic location of the communicating parties. Defensor products are designed to provide Public Key compatibility to non-Entrust® enabled applications using an application independent intercept mechanism.

1. Defensor Client: Generic intercept for TCP/IP, Winsock-based applications – Windows95, 98, Windows NT – Defensor Client is a software for desktop environments. It is designed to secure information exchanged by any Winsock-based application using a single TCP/IP connection, as well as selected multi-connection applications. Defensor Client is used to secure communications at the user level. Because it resides below the applications layers in the protocol stack, Defensor Client provides protection for the data exchanged between a client and the applications.

Applications: FTP, Telnet, Oracle, SQL*Net, Sybase, HTTP, SMTP, POP3 E-mail
2. **Defensor Server:** Manages secure access to Server resources – Solaris 2.6, Windows NT. Defensor Server provides protection by negotiating secure sessions initiated from another Defensor product – either a Client or another Server. It authenticates the user, or application requesting access and then encrypts and protects data integrity as required. Session oriented service provided by Defensor Server can secure applications such as Oracle, Sybase, Telnet, HTTP and FTP.

3. **Defensor Mainframe:** Allows integration of PKI in the mainframe environment – MVS, CICS (IBM). Defensor Mainframe provides both file and session oriented Application Programming Interfaces (APIs) that can secure important information under application control. It allows an organization to extend its public key infrastructure to include mainframe applications. Requires an adjunct processor, which is physically connected to the IBM mainframe using a channel gateway.

4. **Defensor Secure Sign-On:** Provides strong authentication to mainframe applications – MVS/VTAM

An optional extension to Defensor Mainframe capabilities – designed to mitigate risk associated with mainframe identification and authentication information passing in the clear between workstations and the mainframe. Employs PKI-based authentication services to authenticate users to the mainframe.

**Features:**
- Secure end-to-end sessions
- User Authentication: X.509 Digital Certificates and Distinguished Name (DN)
- Certification Authority: Any, as specified in the Digital Certificate
- Entrust/Session V4.0 toolkit provides the cryptographic services

### 5.2 ID2

#### 5.2.1 ID2 CERTIFICATE MANAGER

**Classification:** Certificate Management

Enables Certification Authorities to produce RSA keys and manage X.509 certificates that can be used in a variety of applications requiring secure identification and digital signatures. The product is designed for high-end users such as banks and postal and telecommunication companies.

**Features:**
- RSA encryption keys and digital certificates for smart cards
- Provides RSA keys and certificates for software such as browsers and web servers
- Supports the SEIS and PKCS#15 smart card application profiles.
- Compliant with all PKI standards
- Tamper-resistant hardware board for storage of CA keys
- System monitoring over SNMP
5.2.2 ID2 PERSONAL

Classification: Client Software/ Desktop Security

This is a desktop product that brings security, user authentication and digital signatures to standard Internet browsers by utilizing Smart Cards and the client-side of the SSL protocol.

Features:
- Includes a Cryptographic Service Provider (CSP) for the Microsoft CryptoAPI
- Implements version 2.01 of the PKCS#11 standard
- Smart card reader access using the PC/SC standard
- Supports standard digital signature format – PKCS#7
- Includes full cryptographic support – 128-bit key length
- Uses the standard client authentication procedure in the SSL 3.0 protocol
- Automated configuration of smart card readers
- Utilizes the security proxy feature in standard Internet browsers
- System requirements – Standard browser or a standard mail client
- Platform – Windows NT, Windows 95, Windows 98

5.2.3 ID2 SECURE TRANSPORT

Classification: Client Software

It adds secure authentication in client/server in client/server environments. With the software installed on the client platform, the user can be identified to the corporate network and access applications with a high degree of security.

Features:
- Installs as Winsock 2.0 Layered Service provider (LSP)
- Implements the SSL 3.0 protocol
- Filters traffic on port and on IP address levels
- Includes full cryptographic support – 128-bit key length
- Smart card reader access using the PC/SC standard

5.2.4 ID2 SERVANT

Classification: Server Software – Certificate verification

iD2 Servant prevents the use of revoked certificates and verifies a certificate or digital signature in a signed message. Simplifies and speeds up certificate verification and parsing, thus enhancing overall PKI performance.

Features:
- Parsing and verification of multiple certificates types using simple interfaces
- Parsing and verification of digital signatures
- Can be plugged into iD2 Guardian or Netscape Enterprise Server
- Supports open standards and protocols like LDAP v.2
- Platforms – Windows NT, AIX, Solaris
### 5.2.5 ID2 GUARDIAN

**Classification:** Server Software – SSL and encryption for TCP/IP network

ID2 Guardian adds security and smart card support to Internet/Intranet or other client/server solutions. It operates on multiple platforms and adds standard SSL (Secure Sockets Layer) support and strong 128-bit encryption to any TCP/IP network. Certificates can also be validated by plugging iD2 Servant into iD2 Guardian.

**Features:**
- Provides Internet server security for all major Internet servers
- Provides SSL 3.0 support
- Includes full cryptographic support – 128 bit key length
- Certificate verification
- Platforms – Windows NT, AIX, Solaris

---

### 5.3 BALTIMORE TECHNOLOGIES

#### 5.3.1 UNICERT V3.05

**Classification:** Certificate Authority systems used in PKI

UniCERT is a policy driven Certificate Authority. Its revolutionary Secure Policy Editor and centralized control allows organizations to define all policies within their PKI.

**Features:**
- Supports Certification Authority, Registration Authority (RA), RA gateways like E-mail, Web, and VPN
- Has a PKI and Security Policy Editor
- Supports different certificate types – SSL, S/MIME, IPSEC, SET)
- UniCERT PKI supports smartcard based access control, authenticated messaging and hardware security modules.
- Certificate Management
- Supports X.509 v3 certificates, PKIX messages, DSA, ECDSA, PKCS encryption (RSA) and cryptographic standards, Certificate Revocation Lists (CRLs) via LDAP, Smartcards & tokens
- Digital signatures for authentication and non-repudiation
- Data integrity by message digest and digital signatures
- Designed to scale from small configuration right up to a very large installation distributed across different networks.
- Handles a wide variety of PKI requirements including different certificate types, multiple delivery systems, multiple key support etc.
5.3.2 W/SECURE, X/SECURE, PKI-PLUS

**Classification:** PKI toolkits

The above software development toolkits allow application developers to create secure encrypted sessions between online networked applications. W/Secure contains an implementation of Wireless Transport Layer Security (WTLS). X/Secure is the world’s first commercial product for securing XML systems. It also provides for a complete PKI and XML integration. It allows secure e-business XML systems to be created and supports full PKI and digital signatures for security and non-repudiation. PKI-Plus provides all the necessary functionality for a client to support a Public Key Infrastructure and can work with all standards based Certificate Authorities. Cryptographic algorithms supported include RSA, DES, Triple-DES, IDEA, DSA, Diffie-Hellman and ElGamal. *(Please note that this is not a security product as such, but a software development kit allowing users to develop products that are secure. So this could and probably should be described elsewhere.)*

Other Toolkits from Baltimore Technologies:
Security Protocol Toolkits (SSL and S/MIME): C/SSL, J/SSL, and SMT
Cryptographic toolkit for developing SSL 3.0 and TLS 1.0 secured Internet and Intranet application written in ANSI C and JAVA (C/SSL, J/SSL). Secure Messaging Toolkit (SMT) is a cryptographic toolkit to add full strength S/MIME security to any messaging application.

5.4 XCERT INTERNATIONAL, INC.

**5.4.1 SENTRY CA, SENTRY RA, & WEB SENTRY**

**Classification:** Certificate Management

Sentry CA provides a certificate issuance and management solution with Public Key Infrastructure. It is supposed to be a full-featured certificate authority (CA) solution for large enterprises and public Cas. Another product Sentry Registration Authority (RA) works in conjunction with Sentry CA. The essential function of Sentry RA is to register users into the PKI-enabled application or security infrastructure. It works with the Sentry CA to verify the credentials of certificate requests and to provide certificates to clients. WebSentry is a plug-in module that works with Sentry CA to PKI-enable web servers. It provides high assurance PKI for web servers.

**Features (of Sentry CA):**
- Supports different certificate standards – x509 v3, SSL S/MIME, Ipsec, SET
- Also supports RSA, DSA & ECDSA certificates, PKCS#11 certificates, and Netscape and Microsoft Internet Explorer certificates
- Directory feature supported includes integrated LDAP certificate repository, S/MIME certificate lookup, and certificate status checking via SSL-LDAP
- Cryptographic support: RSA, DSA, ECDSA
- Includes bundled encryption hardware
- Platforms: Window NT, Solaris
For enhanced client security, Sentry CA can deploy certificates on tamper-proof smartcards, for user authentication and workstation independence. Sentry CA can validate certificates on-line during transactions providing better assurance than traditional time-sensitive CRL-based systems.

### 5.4.2 XCERT DEVELOPMENT KIT (XDK)

**Classification:** Software Development Toolkit

XDK enables developers to integrate products with a certificate authority and Public Key Infrastructure (PKI). XDK provides an easy to use, well defined Application Programmer’s Interface (API) that enables programmers to:

- Enroll public keys
- Retrieve certificates
- Retrieve Certificate Revocation Lists (CRLs)
- Check the status of certificates
- Verify certificates

*(Note: Once again this is software development kit for security software. Will have to decide whether to include it in the survey or not.)*

### 5.5 IBM (INTERNATIONAL BUSINESS MACHINES)

#### 5.5.1 SECUREWAY

Description of this software on the IBM web site is a little too non-technical and verbose. Hundreds of lines written but still could not find out what this product is about except that it is supposed to be a *simpler, more secure foundation for e-business*. It supports *directories* using LDAP and is able to integrate VPN policies. What standards are supported, how security is achieved and other technical details are not elaborated upon. Very disappointing.

**Name:** Vault Registry, Payment Registry  
**Company:** International Business Machines (IBM)  
**Classification:** Registration and Certification solution

This is an integrated registration and certification software for business-critical Web applications.  
**Features:**
- Browser based registration requests and administration  
- Audit trail of registration and database certificate processing actions  
- CA key generation and digital signatures  
- Supports CRLs via LDAP or X.500 directory
5.6 CERTICOM

5.6.1 SSL PLUS 2.0

Classification: SSL based networked application

SSL Plus makes it possible to include security protocols without needing an extensive understanding of cryptography and provides a way to build advanced SSL 3.0 security into networked applications. It provides authentication, confidentiality, and message integrity.

Features:
• Supports RSA and Diffie-Hellman Public Key algorithms with 40-bit key length.
• Also supports ECC, DES, Triple DES, RC4, MD5, and SHA-1 cryptographic algorithms
• Elliptic Curve Cryptography for better security on embedded platforms
• ANSI C implementation
• Supports X.509 certificate standard and VeriSign as a Certificate Authority
• Implements SSL protocol versions 2.0 and 3.0

5.6.2 SC-400 SERIES SMART CARD

Classification: Smart Card Authentication

This product with digital signature smart cards is designed to provide identification and authentication for E-commerce transactions (based on ANSI X9.59, SSL, SET, etc.). The SC-400 series digital signature smart cards provide secure, tamper resistant and tamper evident protection of the card-holder’s private key.

Features:
• Digital signatures in less than one second without the need for a cryptographic coprocessor
• Creates full-strength X9.62 standard compliant digital signatures
• Uses ANSI X9.62 Elliptic Curve Digital Signature Algorithm with 163-bit keys. Exceeds RSA/DSA 1024-bit strength
• On card key generation capability
• Additional memory on the card EEPROM to hold a full X.509 certificate or data

5.7 ENTRUST TECHNOLOGIES

5.7.1 ENTRUST/EXPRESS

Classification: Full featured E-mail solution

Entrust’s Express is a complete secure e-mail solution offering key and certificate management, central trust management and automated certificate checks. It enables users to encrypt and digitally sign e-mail using a variety of clients. It is a S/MIME solution and manages security keys.
Features:
• Uses S/MIME and X.509 certificates
• Express automatically checks the status of certificates when sending and receiving messages.
• Automated certificate retrieval as soon as a recipient’s name is typed in the “To:” field making this product easy to use.
• Non-repudiation and message integrity are in-built into Express
• Automated key management is supported by handling key and certificate updates.

5.7.2 ENTRUST/ACCESS, ENTRUST/VPN CONNECTOR

Classification: Secure connections, VPN

It is solutions for transparently and securely connecting remote users to the internal networks. It allows traveling users (like salesman) and telecommuters (home users) to securely share network resources. Entrust/VPN Connector gives remote offices permanent VPN connectivity – LAN-to-LAN or gateway-to-gateway connectivity). It also allows business partners access to common resources via Extranet. The VPN product lines is essential for businesses since it provides flexibility to use the current network resources, allows transparent implementation of security methods for the existing and legacy applications.

Features:
• Two level security with: Entrust/PKI as a Public Key resource and the security of the Internet Protocol Security Standard (IPSec)
• Provides and manages digital IDs for VPN gateways, remote access clients and routers. Digital IDs, once issued, are valid for up to 60 months.
• Easy scaling of the number of remote users
• Supports all VPN devices with PKCS#10 and PKCS#7 standards.

5.7.3 ENTRUST SECURITY (FOR SAP R/3)

Classification: Full security solution for SAP

Entrust Security for SAP R/3 provides user transparent security for electronic data traveling over public and private networks. It is closely integrated with Entrust/PKI and provides secure digital signature and encryption of files and messages and enables SAP R/3 customers to prudently engage in e-business initiatives.

Features:
• Strong authentication and confidentiality for all electronic data
• Data can be encrypted and digitally signed for secure storage and transmission
• Supports Smart Cards (storing of user’s digital ID in security hardware)
• Integrated with Entrust/PKI for security and key management
5.7.4 ENTRUST/PKI

Classification: Public Key Cryptographic System

Entrust/PKI provides a managed PKI for all secure business solutions. It is considered to be a global market and technology leader in managed PKI solutions that enable organizations to conduct e-business securely. It allows people to encrypt, digitally sign, and authenticate electronic transactions across all applications.

Entrust/PKI provides the following capabilities:

- Certification Authority (CA): Supports X.509 certificate standard. Entrust/Authority provides database encryption, integrity checking, CA hardware authorizations.
- Registration Authority (RA): Capability to securely enroll users and allowing for ongoing administration of the users. Provides comprehensive administrative functions.
- Key and Certificate Management: Automatic key update before the old key expires, flexible key specification, automatic key tracking, certificate update and revocation.
- Key Backup and Recovery: Recovery of entire decryption key history.
- Revocation System: Coverage for all revocation formats and standards including CRL Distribution Points, CRL, and OCSP.
- Notarization: Support dual key pairs to provide non-repudiation and key backup.
- PKI Networking: Provides for cross-certification, directory networking support for any LDAP compliant Directory, certificate revocation system networking.
- Policy Management: Supports CA, RA, and user policy management as also a PKI networking policy management.

Comment: Entrust provides a complete deck of security solutions for business applications from Public Key Cryptography to VPN and E-mail.

5.8 RSA SECURITY

RSA stands for Ron Rivest, Adi Shamir, and Leonard Adleman, the three pioneers in encryption and security methods. RSA as a cryptographic methods, developed in 1977, is a public-key cryptosystem that offers both encryption and digital signatures (authentication). RSA Security offer a variety of security products for Electronic Commerce.

5.8.1 RSA AUTHENTICATION SUITE (RSA ACE/SERVER AND RSA SECURID)

Classification: Authentication system

RSA ACE/Server and RSA SecurID solutions provide centralized, strong, two-factor authentication services for enterprise networks and operating systems.
The RSA SecurID family of products uses two-factor user authentication. The complete system consists of three main components: RSA SecurID authenticators which are distributed to the end users, the RSA ACE/Server security server that is used to power RSA SecurID, and RSA ACE/Agent Software that protects specific information resources on the network.

RSA SecurID authenticators are offered in many forms: hardware, software and smart cards. The most common hardware form is the key fob, a device with a built-in chip, an LCD window capable of displaying up to an eight digit number (or tokencode), yet it is small enough to be attached to a key ring. The key fob is initialized with a unique 64-bit seed value; and each minute the internal chip performs an algorithm combining and scrambling the seed value and current time, to create an apparently random number. Other token types include a credit card sized authenticator and the RSA SecurID PINPAD model which requires the entry of the user’s PIN in order to display the tokencode. RSA SecurID software token for Windows duplicates the function of the PINPAD token in the form of a software utility. Software application is also available on the Palm Computing Platform. Another SecurID produce is a smart card with the random seed value stored on it.

Features:
- Key aspects of the system are encrypted and includes PIN, agent-server communications, server database and remote administration sessions
- ACE/Server attempts to detect intrusions or use of stolen tokens
- Authentication with RSA SecurID is enforced before gaining access to the ACE server
- Audit & Reporting: Ace/Server logs all transactions and user activity. Various reporting templates and formats.
- Includes Livingston 2.0 RADIUS server for account management. Compatible with authentication technologies like TACAS+ and Kerberos
- Cryptographic methods used are proprietary

5.8.2 RSA KEON

Classification: Application Security

These PKI-based application security products are a solution for Securing Internet and e-Business
RSA Keon™ 5.0 public key infrastructure (PKI) products enable, manage and simplify the use of digital certificates in today’s leading Internet applications.

RSA Keon Advanced PKI

As companies look to extend their digital certificates across E-business applications, RSA Keon Advanced PKI removes many of the barriers in today’s PKI implementations – interoperability, security and complexity.

Combining the power of the RSA Keon Security Server and RSA Keon Desktop components, the RSA Keon Advanced PKI software is designed to enrich core PKI functionality, supporting digital certificates from any standards-based certificate authority (CA). This new system is designed to help companies freely choose their certificate sources, use a single set of credentials across many applications and be confident that their Internet transactions and communications are secure.
The RSA Keon Security Server provides services for managing users, digital certificate security policies and trust relationships in a public key infrastructure. RSA Keon Certificate Server is a management system that unites a key management engine, a certificate engine, an LDAP certificate repository and a certificate revocation database together into a single package. It provides the trust management foundation for PKI-enabled applications.

The RSA Keon Agent for R/3 is a solution to bring security and benefits of the RSA public key solutions to the leading enterprise resource planning (ERP) software, SAP R/3. This RSA Keon solution, a PKI based product line, secures access to enterprise R/3, providing strong user authentication, application VPN services to protect data across the network, digital signatures, and centralized auditing and reporting. This uses RSA-based public key technology to authenticate the user to the R/3 server thus replacing the reliance on passwords to authenticate users. It requires the successful exchange of public keys with verification using the private key unique to the authorized individual. Authentication in alternate forms – RSA smart cards, hardware token, and software tokens – are also feasible with this software.

Features of Security Server:
- Central management of all users, digital credentials, Certificate Authority trust relationships, CRL verification and management reporting.
- Supports public key standards such as PKCS and PKIX and other standards like LDAP, SSL, and CAPI.
- RSA Keon Security Server is interoperable with LDAP-enabled directories and certificates like those from VeriSign OnSite.
- Allows central policy management
- Provides a common foundation for secure e-business, including e-mail, Web-based applications, VPNs, ERP, distributed database applications, and many others.
Features of RSA Certificate Server:
- Supports SSL and IPSEC certificates
- Supports PKIX and VeriSign Certificate profiles
- Can use personal Certificates for authentication and S/MIME applications
- Has a self-contained LDAP server
- Compatible with leading directory services such as Netscape Directory Server
- Key usage options – either single certificate or separate certificates for digital signature and encryption keys. The second option facilitates non-repudiation.
- Full certificate revocation support – CRL v2, LDIF download format, and LDAP publication
RSA: Products & Relationship with the Certificate Server

Other Products: RSA BSAFE – a security software development kit that allows software and hardware developers to incorporate encryption technologies into their products.

5.9 NETSCAPE

5.9.1 NETSCAPE CERTIFICATE MANAGEMENT SYSTEM

Classification: Certificate Management Systems

Netscape Certificate Management System is one of the more well known Certificate Management systems. It delivers a comprehensive Public-Key Infrastructure (PKI) solution that makes it easy and economical for an enterprise to deploy and manage its own certificate authority. It provides deployment flexibility to issue, renew, suspend, revoke, and manage X.509 v3 certificates. Netscape certificate Management System is a highly scalable certificate-based security solution that enables protection of sensitive data through use of certificates for authentication, web-based form signing, and other security methods that can be broadly deployed on both Extranets and Intranet.

Features:
- Supports certificate requests from clients, servers, and network devices such as Virtual Private Network clients and routers.
• Allows the issue, renewal, suspension, revoking, and management of single and dual-key
certificates.
• Uses modular system design that includes components like Registration Manager, Certificate
Manager, and Data Recover Manager. These components can be run on multiple systems.
• Supports certificate publication and certificate revocation lists (CRLs) publication to any
LDAP v2 or v3 compliant directory.
• Supports X.509v3 certificates, extensions, and certificate revocation lists (CRLs).
• Also supports the following standards – validation: FIPS 140-1, Level3; Digital signing:
PKCS#11; Encryption and Certificates: DSA, RSA, S/MIME, and SSL for communication
between components.
• Supports certificates for network devices such as VPN clients and Cisco routers by issuing
IPSec certificates.
• Integrates with existing security environment through authentication plug-ins for Kerberos and
SecurID.
• Netscape CMS is designed to support Public Key Infrastructure including multi-tier
hierarchies with subordinate certificate authorities.
• Supports signing key lengths of up to 4096 bits and 128-bit SSL encryption.
• Netscape CMS does not support SET certificates and elliptic curve cryptography.

5.9.2 NETSCAPE SECURITY SERVICES

Classification: Secure software development package

Netscape security services is a software development package that enables developers to integrate
the Internet-standard Secure Sockets Layer (SSL) and public-key security capabilities into their
applications. It claims to simplify the development of security-enabled applications that require
strong session encryption and authentication by providing high-level APIs that do not require deep
knowledge of cryptography.

Features:
• Supports SSL protocol for authenticated, encrypted, tamper-proof communication.
• Supports PKCS#11 cryptographic token interface standard for smartcards and hardware
accelerators.
• Supports major public-key, symmetric, and hashing algorithms, including RSA, DES, Triple-
DES, RC2, RC4, RC5, SHA-1, MD2, MD5, DSA
• Does not support Elliptic curve algorithms

Other Netscape products for E-commerce: Netscape Directory Server – a security solution that
provides the infrastructure required to manage user and account information.

5.10 VERISIGN

One of the very successful companies for security products, VeriSign is the world’s largest
Certificate Authority (CA) – a third party authority that is trusted by the community of
certification users. They have a number of products including Firewalls, Certificate Management Systems, security solutions for VPNs, and PKI solutions. The major problem with their product description is the lack of it. It was very difficult to get any technical information about these products. More information has been requested directly from the vendor itself. Meanwhile, here is a overview of some of their products as related to electronic commerce. Products included are: Check Point VPN-1 & SecuRemote – security solutions for VPN; VeriSign OnSite – a Public Key Infrastructure and certificate management system; and VeriSign Digital ID – a digital certificate designed and created by VeriSign.

5.10.1 CHECK POINT VPN-1 & CHECK POINT SECUREMOTE

**Classification:** Security solution for VPN

With a Check Point VPN solution, organizations can build secure communication channels over any IP-based network. Since it supports both LAN-to-LAN and Client-to-LAN topologies, Check Point VPN provides a single solution that meets the needs of corporate networks, remote sites and mobile workers. The SecuRemote software provides secure connectivity to the growing number of remote users with Internet access. Using SecuRemote, Windows 95 and Window NT users can connect to their corporate network via dial-up Internet connection and establish secure VPN sessions to access sensitive network resources.

**Features:**
- Integrated security, traffic control and enterprise management
- It is IPSec compliant
- Supports automated key management
- Support Client-to-LAN and LAN-to-LAN topologies
- Extends VPN to the desktop and laptops
- SecuRemote supports public key infrastructures utilizing X.509 digital certificates and Entrust Certificate Authorities.
- Supports PKCS #11 standards for software and hardware tokens.

5.10.2 VERISIGN DIGITAL ID

**Classification:** Digital Certificate standard

VeriSign has developed its own digital certificate called a Digital ID. VeriSign’s Digital ID is an encrypted data file that includes:
- The name of the holder and other identification information, such as e-mail address
- A public key, which can be used to verify the digital signature of a message sender previously signed with the matching mathematically unique private key
- The name of the issuer, or Certificate Authority
- The certificate’s validity period

All this information is digitally signed and sealed by the CA and can be verified by anyone. In addition to all the benefits of authentication Digital IDs provide for electronic commerce, Digital IDs also make access control more secure and easier to administer than traditional password
5.10.3 VERISIGN ONSITE (PKI)

Classification: Public Key Infrastructure / CMS

VeriSign OnSite 4.0 is the first integrated PKI platform to support a wide range of Intranet, Extranet, VPN, and E-commerce applications.

Features:
• Key management and recovery
• Automated Certificate Validation and Renewal
• Supports VPN
• Supports dual key cryptography
• Supports the following standards: S/MIME, SSL, and IPsec

Unfortunately there is not much information available on this important product. More information has been requested from the company itself. But until then, this all that could be got from the Internet.

5.11 ENTEGRITY SOLUTIONS

Entegrity Solutions has been very successful in marketing their security products for Electronic commerce. They have a variety of product from e-mail to certificate management systems. Reviewed here are three of their products: The Notary – a certificate management system, SDP (Security Development Platform) – a PKI-enabled software development kit, and AssureWeb – a security product for safe e-commerce transaction on the Worldwide Web.

5.11.1 NOTARY

Classification: Certification Management System (PKI)

Notary from Entegrity Solutions is a complete public-key infrastructure (PKI) certification management system. Notary enables businesses and end users to create, sign, distribute, verify, use, and revoke digital certificates and their corresponding public/private key pairs.

Cas are typically organized in a hierarchy of trust and authority containing several levels. The highest level CA certifies the next level below. The lowest level of CA is usually located within a workgroup or department of the enterprise. Notary can support a wide range of CA topologies and cross-certifications. User registration functionality is also built into Notary. Notary supports tokens such as floppy diskettes and cryptographic smart cards that are used to protect the owner’s private key thus providing a stronger level of authentication than with prevailing systems.

Entegrity Solutions Notary software provides the basis for implementing an information security
framework in an IT infrastructure. Notary provides the digital certificate infrastructure used by software applications developed using Entegrity’s Security Development Platform (SDP).

Features:

- Supports centralized policy management to the certification environment
- Supports X.509 v3 extensions and v2 CRL (certificate revocation list)
- Supports certificate management protocol messages: PKCS #7, PKCS #10, and PKIX
- Supports RSA, Diffie-Hellman, and DSA algorithms
- Generates user keys at server or at client system
- Supports key backup and key recovery
- Supports split keys: one for encryption and one for signing
- Enables users to store private key pairs on De La Rue, Gemplus, or Siemens smart cards
- Supports LDAP v2 for certificate and CRL distribution
- Supports CRL distribution points for quick uniform access to CRLs
- Supports cross-certification between separate Notary domains as well as other industry-leading CAs which follow the PKIX standards
- Supports strong cryptographic algorithms including Diffie-Hellman, RSA, IDEA, and Blowfish.

The Notary Management System (NMS) software provides the capability to issue, personalize, and use cryptographic smart cards or security diskettes. It provides the capability to issue, revoke, and distribute certificates and to issue certificate revocation lists (CRLs). Notary Client software allows users to request certification and generate their own public/private key pairs, as well as to retrieve other people’s certificates from an LDAP-compliant directory server and verify them. Adjustable security policy settings software module allows the security administrator to set the environment of the certification authority to match the organization’s security policy.

5.11.3 SDP – SECURITY DEVELOPMENT PLATFORM

Classification: PKI-based Software Development

Security Development Platform (SDP) from Entegrity Solutions is a software development environment that enables enterprises to deploy public-key infrastructure-based (PKI-based) security solutions into the enterprise. Using SDP, an enterprise can rapidly enhance or develop applications for complete public-key infrastructure (PKI) support-including protocols for digital certification management for clients and servers; encryption-including digital signatures, encryption/decryption of information, and messaging integrity; user authentication-including token smart cards or floppy disks support and bilateral or mutual authentication using digital certificates.

Entegrity’s SDP can be viewed as a class framework consisting of the Public key infrastructure (PKI) components:

- Certificate services, Policy and personal security
- environment (PSE), X.509 v3 and LDAP v2
- components, PKI database and distribution
- Cryptographic mechanisms
- Token system (TOSY)
- Messaging and message transport components

Features:

- Uses object-oriented platform with C++ APIs
- Supports X.509 v3 certificates and v2 CRLs
- Supports LDAP v2 for certificate and CRL distribution
- Supports PKIX, PKCS #7, and PKCS #10 for certificate issuance
- Cryptography with up to 2048-bit key lengths
- Digital Signatures: DSA (SHA-1) standard; RSA (SHA-1), RSA (MD2), and RSA (MD5) optional
- Supports multiple token technologies including: Smart cards and smart card readers from De La Rue, Gemplus, Siemens, and Philips Floppy and hard disk tokens

5.11.4 ASSUREWEB

Classification: Security product for Worldwide Web transactions

AssureWeb from Entegrity Solutions provides enterprises with a set of services to enable a secure environment for conducting e-commerce business transactions with non-repudiation over the World Wide Web (WWW). Through the use of strong public-key cryptography (PKC) and digital certification technology, Entegrity AssureWeb provides strong authentication, access control, and document protection in Internet, intranet, or extranet environments.

AssureWeb works in conjunction with industry-leading Web servers from Netscape, Microsoft, Apache, and others to provide a secure Web environment for internal applications, electronic commerce, and business-to-business transactions. AssureWeb supports security “tokens” (hardware tokens such as smart cards and software tokens that can be stored on floppy diskettes) which provide an even higher level of security and confidence for electronic business transactions in the global network environment. AssureWeb uses the digital certificates generated by a certification authority (CA) such as Entegrity’s Notary CA.

Features:

- Standards supported: Public-key infrastructure X.509 v3 certificates and v2 CRLs (certificate revocation lists) S/MIME
- Certificate management protocols: PKCS #7, PKCS #10, PKIX, and LDAP v2. True fine-grained access control (FGAC) to Web pages and Web objects based on the use of strong digital certification
- Web servers support: Netscape Web servers, Microsoft Internet Information Server (IIS), Apache Web server, or any general purpose Web server
- Authentication: client and Web site mutual authentication for extranet applications based on
• Certificates and tokens  Data privacy using strong cryptography (e.g. RSA, DES, Triple DES, IDEA, and Blowfish)
• Integrity and non-repudiation through the use of MACs and digital signatures

5.12 KERBEROS

Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography system such as DES. The Kerberos protocol uses cryptography so that a client can prove its identity to a server (and vice versa) across an insecure network connection. After a client and server has used Kerberos to prove their identity, they can also encrypt all of their communications to assure privacy (preventing unauthorized reading) and data integrity (detection of modification) as they go about their business.

Kerberos is a network security system which relies on cryptographic methods for its security. Since Kerberos' encryption system, DES, is not exportable, Kerberos itself cannot be exported or used outside of the United States in its original form. As a partial solution to this problem, the Kerberos source code was modified by the addition of #ifdef NOENCRYPTION around all calls to DES functions. Compiling this version with the symbol NOENCRYPTION defined results in a system that looks like Kerberos from an application's point of view but that does not require DES libraries (and, as a result, does not speak the real Kerberos protocol and does not provide any security).

Kerberos was developed in the mid-'80s as part of MIT's Project Athena [2]. As use of Kerberos spread to other environments, changes were needed to support new policies and patterns of use. To address these needs, design of Version 5 of Kerberos (V5) began in 1989 [11]. Though V4 still runs at many sites, V5 is considered to be standard Kerberos.
Limitations of Kerberos

In particular, Kerberos is not effective against password guessing attacks; if a user chooses a poor password, then an attacker guessing that password can impersonate the user. Similarly, Kerberos requires a trusted path through which passwords are entered. If the user enters a password to a program that has already been modified by an attacker (a Trojan horse), or if the path between the user and the initial authentication program can be monitored, then an attacker may obtain sufficient information to impersonate the user. Kerberos can be combined with other techniques, as described later, to address these limitations.

To be useful, Kerberos must be integrated with other parts of the system. It does not protect all messages sent between two computers; it only protects the messages from software that has been written or modified to use it. While it may be used to exchange encryption keys when establishing link encryption and network level security services, this would require changes to the network software of the hosts involved.

Kerberos does not itself provide authorization, but V5 Kerberos passes authorization information generated by other services. In this manner, Kerberos can be used as a base for building separate distributed authorization services.

How Kerberos works

The Kerberos Authentication System uses a series of encrypted messages to prove to a verifier that a client is running on behalf of a particular user. The Kerberos protocol is based in part on the Needham and Schroeder authentication protocol, but with changes to support the needs of the environment for which it was developed.
Among these changes are the use of timestamps to reduce the number of messages needed for basic authentication, the addition of a "ticket-granting" service to support subsequent authentication without re-entry of a principal's password, and different approach to cross realm authentication (authentication of a principal registered with a different authentication server than the verifier).

### 5.13 PRODUCT COMPARISON

<table>
<thead>
<tr>
<th>Feature</th>
<th>ID2</th>
<th>Baltimore Technologies</th>
<th>Unicert</th>
<th>Xcert Sentry</th>
<th>Entrust</th>
<th>RSA KEON</th>
<th>Netscape CMS</th>
<th>VeriSign</th>
<th>Entregrity Notary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Card support?</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Hardware key support</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>VPN support</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Certificate Revocation</td>
<td>?</td>
<td>CRL</td>
<td>CRL</td>
<td>CRL, OCSP</td>
<td>CRL</td>
<td>CRL</td>
<td>CRL</td>
<td>CRL</td>
<td>CRL</td>
</tr>
<tr>
<td>Encryption and</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X.509</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>PKCS#11</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>SSL</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>DSA</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>ECDSA</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>RSA</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Diffie-Hellman</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>S/MIME</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>SET</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Others</td>
<td>PKCS#7, IPSEC, SET</td>
<td>IPSEC, Netscape, Microsoft IE</td>
<td>IPSec</td>
<td>IPSec</td>
<td>IPSec</td>
<td></td>
<td>PKIX, PKCS#10, PKCS#7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cross -certification</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>key backup and recovery</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>LDAP support</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Elliptic curve</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Signing key</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Encryption key length</td>
<td>128</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dual key/split key (non-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>repudiation)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Product Comparison for PKI/ Certificate Management Systems
REFERENCES

5. “Netscape, others roll out security wares at RSA show”, INFOWORLD JANUARY 18,1999