

Security Threats

There are numerous threats to security of applications and data. With the increasing use of internet and the advancing IT, applications are becoming increasingly vulnerable to threats that could be a malicious code, viruses, worms, etc.

Some of the security threats are as follows:-

Virus Attack

A computer virus is a man-made program or piece of code that is loaded onto one's computer without the victims' knowledge and runs against his/her wishes.

Viruses can also replicate themselves over and over again and is relatively easy to produce. Even a simple virus is dangerous because it corrupts the system.

An even more dangerous type of virus is the one capable of transmitting itself across networks and bypassing security systems.

Viruses can be transmitted as attachments to an e-mail note or in a downloaded file, or be present on a diskette or CD.

The sender of the e-mail note, downloaded file, or diskette you've received is usually unaware that it contains a virus. Some viruses wreak havoc as soon as their code is executed while other viruses lie dormant until circumstances cause their code to be executed by the computer.

E-mail viruses:

An e-mail virus travels as an attachment to e-mail messages, and usually replicates itself by automatically mailing itself to dozens of people in the victim's e-mail address book. Some e-mail viruses don't even require a double-click they launch when you view the infected message in the preview pane of your e-mail software.

The different damages a virus can cause:

- An annoying message appearing on the computer screen.
- Reduce memory or disk space.
- Modify existing data.
- Overwrite or Damage files.
- Erase hard drive.

PROTECTION TIPS:

- Use anti-virus from good brands like Mc-Afee or Kaspersky.
- Turn on 'auto update' option for your browser and plug-ins.
- Install Anti-malware.
- For extra security, run anti-malwares by different brands.
- Set a strong password for your FTP.
- Configure FTP client settings. Activate the option to "Always use SFTP".
- Avoid sites that do not look trustworthy.
- Avoid sites in which 'https' is clearly striked out.
- Quick Scan pen drives and flash drives when you insert them into your systems.
- Scan your systems frequently.

UNIT-II: CYBER SECURITY

Worm

Computer worms are standalone malware programs that will use your computer network to replicate themselves in order to spread to other computers. Unlike a computer virus, it does not need to attach itself to any program, file or document.

In some ways worms are more deadly than viruses because they don't need to lodge themselves into programs to replicate. Worms can replicate independently through your system.

Once in your system, worms will look scan your network for other machines that may have similar security holes. If the worm finds one, it will copy itself into the new computer and start the process all over again.

Worms use parts of an operating system that are automatic and usually invisible to the user. It is common for worms to be noticed only when their uncontrolled replication consumes system resources, slowing or halting other tasks. Worms can perform a variety of operations according to how it has been designed.

- It can cause a denial of service attack
- It gets attached to Microsoft outlook or any such mailing facility and sends mails to everybody on the address list (replicates itself and passes on the worm to everyone in the address list),
- overwrites your files and documents, and
- Makes your computer slow and dis-functional.

Illustration:

The ILOVEYOU virus comes in an e-mail note with "I LOVE YOU" in the subject line and contains an attachment that, when opened, results in the message being re-sent to everyone in the recipient's Microsoft Outlook address book and, perhaps more seriously, the loss of every JPEG, MP3, and certain other files on the recipient's hard disk.

As Microsoft Outlook is widely installed as the e-mail handler in corporate networks, the ILOVEYOU virus can spread rapidly from user to user within a corporation. On May 4, 2000, the virus spread so quickly that e-mail had to be shut down in a number of major enterprises such as the Ford Motor Company. The virus reached an estimated 45 million users in a single day.

PROTECTION TIPS:

- Use anti-virus from good brands like Mc-Afee or Kaspersky.
- Turn on 'auto update' option for your browser and plug-ins.
- Install Anti- malware.
- For extra security, run anti-malwares by different brands.
- Set a strong password for your FTP.
- Configure FTP client settings. Activate the option to "Always use SFTP".
- Avoid sites that do not look trustworthy.
- Avoid sites in which 'https' is clearly removed.
- Scan pen drives and flash drives when you insert them into your systems.
- Scan your systems frequently.

UNIT-II: CYBER SECURITY

Trojan

In the 12th century BC, Greece declared war on the city of Troy. The dispute was caused due to the fact that the prince of Troy and the Queen of Sparta eloped. Hence declaring that they intend to marry.

The Greeks besieged Troy for 10 years but met with no success as Troy was very well fortified.

In a last effort, the Greek army pretended to be retreating, and left behind a huge wooden horse. The people of Troy saw the horse and thought it was a gift from the Greeks.

They pulled the horse into their city, unaware that the hollow wooden horse had some of the best Greek soldiers hiding inside it.

Under the cover of night, the soldiers snuck out and opened the gates of the city, and later, together with the rest of the army, besieged and destroyed Troy. Similar to the wooden horse, a Computer Trojan (also referred to as Trojan Horse program) pretends to do one thing while actually doing something completely different.

A Trojan horse program is a program that appears to have some useful or benign purpose, but really masks some hidden malicious functionality.

Today's Trojan horses try to sneak past computer security fortifications (such as firewalls), by employing like-minded trickery. By looking like normal software, Trojan horse programs are used for the following goals:

- Duping a user or system administrator into installing the Trojan horse in the first place. In this case, the Trojan horse and the unsuspecting user becomes the entry vehicle for the malicious software on the system.
- Blending in with the “normal” programs running on a machine. The Trojan horse camouflages itself to appear to belong on the system so users and administrators continue their activity, unaware of the malicious code's presence.

Attackers have devised a myriad of methods for hiding malicious capabilities inside their wares on your computer. These techniques include

- employing simple, yet highly effective naming games,
- using executable wrappers,
- attacking software distribution sites,
- manipulating source code,
- co-opting software installed on your system, and
- disguising items using polymorphic coding techniques.

As we discuss each of these elements, we must bear in mind that the attackers' main goal is to disguise the malicious code so that the victims do not realize what the attacker is up to.

Types of Trojans

The most common types of Trojans found today are:

1. Remote Administration Trojans (RATs)

These are the most popular Trojans. They let a hacker access the victim's hard disk, and also perform many functions on his computer (shut down his computer, open and shut his CD-ROM drive etc.).

UNIT-II: CYBER SECURITY

Modern RATs are very simple to use. They come packaged with two files - the server file and the client file. The hacker tricks someone into running the server file, gets his IP address and gets full control over the victim computer.

Some Trojans are limited by their functions, but more functions also mean larger server files. Some Trojans are merely meant for the attacker to use them to upload another Trojan to the target's computer and run it; hence they take very little disk space. Hackers also bind Trojans into other programs, which appear to be legitimate, e.g. a RAT could be bound with an e-greeting card.

Most RATs are used for malicious purposes - to irritate or scare people or harm computers. There are many programs that detect common Trojans. Firewalls and anti-virus software can be useful in tracing RATs.

RATs open a port on your computer and bind themselves to it (make the server file listen to incoming connections and data going through these ports). Then, once someone runs his client program and enters the victim's IP address, the Trojan starts receiving commands from the attacker and runs them on the victim's computer.

Some Trojans let the hacker change this port into any other port and also put a password so only the person who infects the specific computer will be able to use the Trojan. In some cases the creator of the Trojan would also put a backdoor within the server file itself so he'll be able to access any computer running his Trojan without the need to enter a password.

This is called "a backdoor within a backdoor" e.g. CIA, Netbus, Back Orifice, Sub7.

2. Password Trojans

Password Trojans search the victim's computer for passwords and then send them to the attacker or the author of the Trojan. Whether it's an Internet password or an email password there is a Trojan for every password. These Trojans usually send the information back to the attacker via email.

3. Privileges-Elevating Trojans

These Trojans are usually used to fool system administrators. They can either be bound into a common system utility or pretend to be something harmless and even quite useful and appealing. Once the administrator runs it, the Trojan will give the attacker more privileges on the system. These Trojans can also be sent to less-privileged users and give the attacker access to their account.

4. Key loggers

These Trojans are very simple. They log all of the victim's keystrokes on the keyboard (including passwords), and then either save them on a file or email them to the attacker once in a while. Key loggers usually don't take much disk space and can masquerade as important utilities, thus becoming very hard to detect.

5. Joke Programs

Joke programs are not harmful. They can either pretend to be formatting your hard drive, sending all of your passwords to some hacker, turning in all information about illegal and pirated software you might have on your computer to the police etc. In reality, these programs do not do anything.

6. Destructive Trojans

These Trojans can destroy the victim's entire hard drive, encrypt or just scramble important files. Some might seem like joke programs, while they are actually destroying every file they encounter. In an unreported case in India, a Trojan almost led to the death of a reporter!

Logic Bomb

A logic bomb is a piece of code intentionally inserted into a software system which when triggered will set off a malicious task such as reformatting, and/or deleting, altering or corrupting data on a hard drive. It's secretly inserted into the code of a computer's existing software, where it lies dormant until that event occurs.

A program in which damage is delivered when a particular logical condition occurs; e.g., not having the author's name in the payroll file. Logic bombs are a kind of Trojan Horse and most viruses are logic bombs.

PROTECTION TIPS:

- Always change passwords frequently. They save users from a lot of trouble.
- Use security measures to detect insider threats in your system. Basic anti-viruses are not efficient enough.

Phishing & Spoofing attacks

In the 19th century, British comedian Arthur Roberts invented a game called Spoof, which involved trickery and nonsense. This gave the English speaking world a new word that today symbolizes a gamut of hacking technologies.

Spoofing attacks primarily include e-mail spoofing, SMS spoofing, IP spoofing, and web spoofing. Spoofing attacks are used to trick people into divulging confidential information (e.g. credit card data) or doing something that they would usually not do (e.g. installing malicious software on their own computers).

Such use of spoofing attacks is commonly referred to as Phishing.

Sending an e-mail from somebody else's e-mail ID is the simplest form of **Email spoofing**. Innumerable tools exist on the Internet which can easily be used to send e-mails appearing to have been sent by somebody else. The effects are intense.

Case: Many customers received an email from their bank asking them to verify their usernames and passwords for the bank records. The emails were spoofed, but thousands of customers clicked on the link in the email and submitted the information at the webpage that opened up. On investigation, it is found that the emails were sent by a disgruntled employee.

Case: Thousands of employees of a global IT company ended up installing viruses on their computers when they executed an attachment appearing to have been sent out by their officers. The employees even disabled the anti-virus software because the email said that "the attachment may be incorrectly detected as a virus!" On investigation, it was found that the emails had been sent out by a rival company.

UNIT-II: CYBER SECURITY

SMS spoofing is very similar to e-mail spoofing. The major difference being that instead of an email ID, a cell phone number is spoofed and instead of a spoofed e-mail, a spoofed SMS is sent.

Case: A young lady received an SMS from her husband's cell phone informing her that he had had an accident and was at the hospital and urgently needed money. On receiving the SMS, she rushed out of the house with the money. She was attacked and robbed by the person who had sent her the spoofed SMS.

An IP address (e.g. 75.125.232.93) is the primary identification of a computer connected to a network (e.g. the Internet). A criminal usually uses IP spoofing to bypass IP based authentication or to mislead investigators by leaving a trail of false evidence. IP spoofing can be accomplished using proxy servers and simple PHP scripts that are readily and freely available online.

Case: Internet users in many countries use proxy servers to bypass Government imposed Internet censorship. (We are not passing any comment on whether is it right or wrong to impose Internet censorship or bypass it, as the case may be.)

DNS spoofing involves manipulating the domain name system to take unsuspecting victims to fake websites (that look identical to the original ones). Sitting at the computer you may type in www.asianlaws.org but the site that opens up may be a fake site!

This can and has been done at the local organizational level (e.g. by host file rewriting or by a network administrator with malicious intentions) or at the national or international level (by hackers exploiting vulnerabilities in the BIND software that runs most of the world's domain name servers).

Case: Hundreds of employees at a global financial services company received emails from a popular online store about a huge discount on some popular books and DVDs. On clicking the link in the email, users were taken to what appeared to be the website of the online store. Most of the recipients of the emails placed orders using their credit cards. No one got the books or the DVDs, all got was a hefty credit card bills at the end of the month.

On investigation it was uncovered that the network administrators had connived to carry out a simple Phishing attack. It was a fake email and a fake website. None of the victims (most of whom were advanced computer users) realized that something was amiss.

PROTECTION TIPS:

- Enable authentication based on the key exchange on your network. IPsec will significantly reduce the risk of spoofing.
- Ensure you use access control to deny private IP addresses on your downstream interface.
- Filter inbound and outbound traffic.
- Preferably, in cases of suspicion, always ensure if the sender actually sent the mail or sms.

UNIT-II: CYBER SECURITY

Malware (Malicious Software)

Malware, short for malicious software, is software used or created by hackers to infiltrate or damage or disrupt computer operation, gather sensitive information, or gain access to private computer systems. While it is often software, it can also appear in the form of scripts or code. 'Malware' is a general term used to refer to a variety of forms of hostile, intrusive, or annoying software.

Malware includes computer, worms, trojan horses, spyware, adware, most rootkits, and other malicious programs. In law, malware is sometimes known as a computer contaminant, as in the legal codes of several U.S. states. Some malware is disguised as genuine software, and may come from an official company website.

Malware is used primarily to steal sensitive personal, financial, or business information for the benefit of others. It can also hijack your browser, redirect your search attempts, serve up nasty pop-up ads, track what web sites you visit etc. Malware is sometimes used broadly against corporations to gather guarded information, but also to disrupt their operation in general. Many malwares will reinstall themselves even after you think you have removed them, or hide themselves deep within Windows, making them very difficult to clean.

Left un-guarded, personal and networked computers can be at considerable risk against malware threats.

As per an analysis done in the Kaspersky lab, the following is the result.

PROTECTION TIPS:

- Use a firewall.
- Keep track and control your emails.
- Use up-to date antivirus software.
- Scan your computer frequently to detect unknown malicious programs running.
- Secure your browser.

SECURITY THREATS TO E-COMMERCE

Internet, being a public domain, is open to all. Each and every transaction that occurs on the internet can be tracked, monitored logged, and stored. The information is shared over the internet or carrying out transactions is constantly under security threats. These threats may originate from internal or external sources. Thus, it becomes imperative for businesses to understand these security threats well before making their presence online. Some of the top security threats from internal and external sources are as follows:-

1. Unauthorized internal users who may access confidential information by using stolen passwords for committing fraud or theft.
2. Former employees of an organization who have maintained access to the information sources directly by creating alternative passwords, “back doors” into the computer system, or indirectly through former co-workers.
3. Weak access points in information infrastructure and security that can expose company information and trade secrets.
4. Management that undermines security may be the greatest risk to e-commerce.

UNIT-II: CYBER SECURITY

5. Contractors, partners, consultants, etc. who take benefit of even limited access to important systems.

An increase in the sale of various anti-virus software and subscription to e-mail virus protection indicate that people are increasingly becoming aware of these threats. Businesses can make it mandatory for their service providers and merchants to have firewalls, encryption, as well as testing and access policies as a condition of doing business with them.

E-CASH AND ELECTRONIC PAYMENT SYSTEM

ELECTRONIC PAYMENT SYSTEM

Electronic payments are the central part of E-Commerce activities as it deals with the strategies for the payment of goods and services by online customers. Electronic payments also refer to the activity of account settlement where the prompt settlement of payments is crucial. If the debit and credit to the bank account, customer and the company are not settled immediately or suffers due to conventional processing delays, then the entire business chain may be interrupted. Payment and settlement of the business chain may be interrupted. Payment and settlement of the business account are the bottleneck in all E-Commerce activities. Conventional instruments for payments such as demand draft, credit notes, and cheques are not suited to E-Commerce. The electronic version of this instrument also may not work well particularly when small payments are to be made. The supplier as well as the customer would like to settle the payment online when the amount to be paid is small. Conventional instruments are too slow to be processed and the overheads in processing such instruments may be high.

Electronic Payment is defined as, “Electronic Payment is a financial exchange that takes place online between buyers and sellers. The content of this exchange is usually some form of digital financial instrument (such as encrypted credit card numbers, electronic cheques or digital cash) that is backed by a bank or an intermediary, or by a legal tender.”

Various instruments which may be used to make payment on the Internet are Credit/Debit cards, Smart Cards, Electronic Cash, Electronic Wallet etc. Important issues related to the Electronic payment system include the methods, form and characteristic of payment instrument such as credit/debit cards, how to minimize the financial risk such as leakage of information, mistakes and frauds and finally devising methods for the completion of electronic payment cycle.

Several protocols have been devised and deployed to provide the necessary security to payment transactions. The Netscape Navigator and MS Internet Explorer have promoted their payment protocols for safe payment transaction and these have been implemented in their web browsers. Some of the commonly used protocols for secured transaction are as follows:

1. Secure Electronic Payment Protocol (SEPP)
2. Secure Electronic Transaction Protocol (SETP)

UNIT-II: CYBER SECURITY

Secure Electronic Payment Protocol (SEPP)

SEPP has been developed by Master Card and Netscape. It has been implemented in the Netscape web browsers. Secure Electronic Transaction Protocol is a new E-Commerce industry Standard. Many companies are implementing SET protocol and it has many features in common with SEPP. SET is also one of the popular protocols for safe electronic transaction. It was developed by Microsoft and VISA. SET uses the cryptographic standard and digital signatures to secure the authentication and verification is done using digital signatures.

Electronic Payments through SEPP Protocol:

This protocol was developed jointly by Master Cards, e-Cash, Netscape and IBM. SEPP is an open vendor free and license free specification for online payment transactions. SEPP works on the simple conventional process of signing and submission. This protocol involves three way communications involving customer, company and the bank. The process is shown in the fig(1).

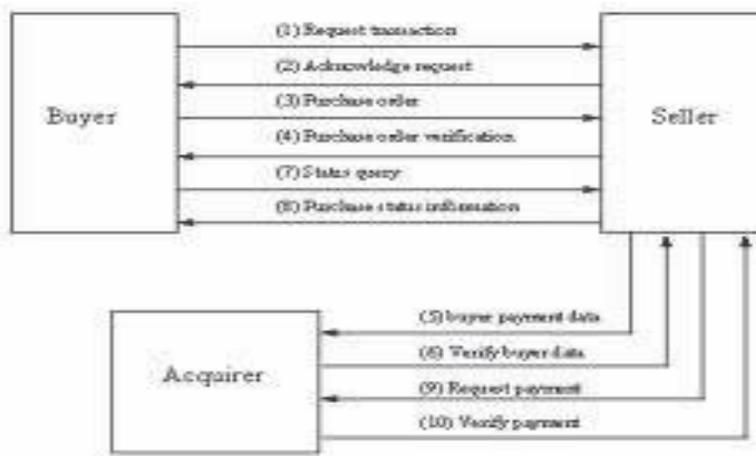


Fig: Electronic Payment System

As shown in the figure, the customer is a person who buys the product or services from a company. He does it by visiting the web site of the company. The customer is also a holder of a credit card and intends to pay for the purchased product and services through his credit card.

The company's web site displays and sells the items such as goods and services. It also accepts the payment from the customer against the purchased items and services.

Company's Banker is essentially a bank that services the company's account and processes credit card based transaction for the company.

Certificate management system represents the agent or broker who creates and distributes the digital certificate to customers and other financial institutions. The agent works on behalf of the credit card issuer bank. The whole process of purchase is done in the following steps:

UNIT-II: CYBER SECURITY

1. The customer sends an initial message to the company web site.
2. Company response by sending the invoice message, enabling the customer to validate goods and services of the company.
3. The customer then prepares the purchase order the credit cards details. The credit card details are so encrypted that can only be decrypted by the Bank.
4. The company receives the purchase order. The company then sends the encrypted credit card details to its banker.
5. The banker decrypts the credit card details and may verify from the credit issuer bank.
6. When the company's banker verifies the authenticity of the credit card, is send an authorization response essentially contains the verification results of credit card.
7. The Company then sends to the customer a purchase order response.
8. Next step is the delivery of the product to the customer. The non-tangible items are delivered to the customer immediately and if the purchase order contains the delivery of a tangible item, the some are shipped by conventional means.

The various factors that have leaded the financial institutions to make use of electronic payments are:

1. Decreasing technology cost:

The technology used in the networks is decreasing day by day, which is evident from the fact that computers are now dirt-cheap and Internet is becoming free almost everywhere in the world.

2. Reduced operational and processing cost:

Due to reduced technology cost the processing cost of various commerce activities becomes very less. A very simple reason to prove this is the fact that in electronic transactions we save both paper and time.

3. Increasing online commerce:

a. Affecting the consumers:

Credit cards, Debit Cards, ATMs (Automated Teller Machines), stored value cards, E-Banking.

b. Enabling online commerce:

Digital Cash, E-Cash, Smart cards (or Electronic Purse) and encrypted Credit cards.

TYPES OF ELECTRONIC PAYMENT SYSTEM:

Electronic payment systems are proliferating in banking, retail, health care, on-line markets. Organizations are motivated by the need to deliver products and service more cost effectively and to provide a higher quality of service to customers. The emerging electronic payment technology was labeled Electronic Funds Transfer (EFT). EFT is defined as any transfer of funds initiated through an electronic terminal, telephonic instrument or computer or magnetic tape so as to order, instructor authorize a financial institution

UNIT-II: CYBER SECURITY

to debit or credit an account. EFT utilizes computer and telecommunication components both to supply and to transfer money or financial assets. Transfer is information-based and intangible. Thus, EFT stands in marked contrast to conventional money and payment modes that rely on physical delivery of cash or checks by truck, train or airplane. Work on EFT can be segmented into three broad categories:

1. Banking and financial payments

- Large-scale or wholesale payments (e.g.: Bank-to-Bank transfer)
- Small-scale or retail payments (e.g.: Automated teller machines and cash-dispensers)
- Home banking (e.g.: Bill payment)

2. Retailing payments

- Credit cards (e.g.: VISA or Master cards)
- Private label credit/debit cards
- Change cards

3. On-line electronic commerce payments

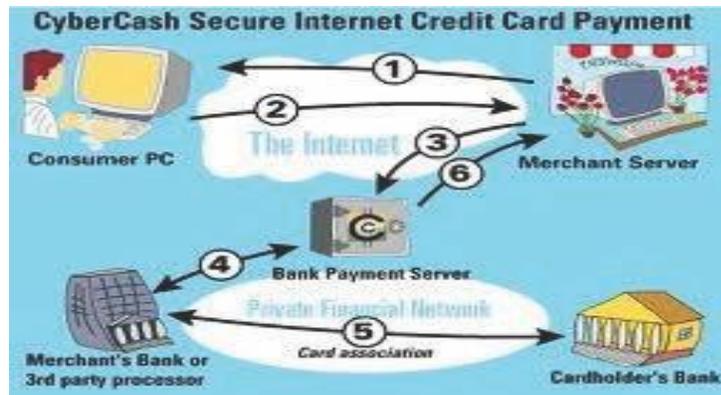
- Token- based payment system
 - ➔ Electronic cash (e.g.: Digicash)
 - ➔ Electronic checks (e.g.: Netcheque)
 - ➔ Smart cards or debit cards
- **Credit card-based payment system**
 - ➔ Encrypted credit cards (eg.: www from-based encryption)
 - ➔ Third-party authorization numbers

CREDIT CARDS

Credit card is working on the *postpaid* mechanism. Credit cards are another payment instrument which has now become very common. Credit cards are issued by a financial institution which allows you to make purchases up to a certain limit on credit. Most of the credit card companies recognize the organization or shop etc. from where you may purchase the item. Payment of these items is made by the credit card company or your behalf. The credit card companies regularly send the bill to the customer for the customers for the shopping they have done. In E-Commerce, use of credit card is very common. If consumers want to purchase a product or service, they simply send their credit card details to the service provider involved and the credit card organization will handle this payment like any other transaction.

Figure shows the various steps when items are purchased on the Internet through Credit cards.

UNIT-II: CYBER SECURITY



The figure shows the processing of the credit card when an item is purchased from a company or the web. When the customer wishes to purchase items from a web site, he places the purchase order electronically and sends encrypted credit card number. This information will be sent to the customer bank through the credit card processor. After checking the authenticity of the credit card, the banks allow the company to go ahead with the purchase. The bank will issue an electronic token to the company. The customer bank will realize the payment through monthly or fortnightly bill sent to the customer.

ADVANTAGES AND DISADVANTAGES TO CREDIT CARDS

ADVANTAGES:

- **Purchase Power and Ease of Purchase** - Credit cards can make it easier to buy things. If you don't like to carry large amounts of cash with you or if a company doesn't accept cash purchases (for example most airlines, hotels, and car rental agencies), putting purchases on a credit card can make buying things easier.
- **Protection of Purchases** - Credit cards may also offer you additional protection if something you have bought is lost, damaged, or stolen. Both your credit card statement (and the credit card company) can vouch for the fact that you have made a purchase if the original receipt is lost or stolen. In addition, some credit card companies offer insurance on large purchases.
- **Building a Credit Line** - Having a good credit history is often important, not only when applying for credit cards, but also when applying for things such as loans, rental applications, or even some jobs. Having a credit card and using it wisely (making payments on time and in full each month) will help you build a good credit history.
- **Emergencies** - Credit cards can also be useful in times of emergency. While you should avoid spending outside your budget (or money you don't have!), sometimes emergencies (such as your car breaking down or flood or fire) may lead to a large purchase (like the need for a rental car or a motel room for several nights.)
- **Credit Card Benefits** - In addition to the benefits listed above, some credit cards offer additional benefits, such as discounts from particular stores or companies, bonuses such as free airline miles or travel discounts, and special insurances (like travel or life insurance.)

UNIT-II: CYBER SECURITY

While most of these benefits are meant to encourage you to charge more money on your credit card (remember, credit card companies start making their money when you can't afford to pay off your charges!) the benefits are real and can be helpful as long as you remember your spending limits.

DISADVANTAGES:

- **Blowing Your Budget --** The biggest disadvantage of credit cards is that they encourage people to spend money that they don't have. Most credit cards do not require you to pay off your balance each month, so even if you only have \$100, you may be able to spend up to \$500 or \$1,000 on your credit card. While this may seem like 'free money' at the time, you will have to pay it off -- and the longer you wait, the more money you will owe since credit card companies charge you interest each month on the money you have borrowed.
- **High Interest Rates and Increased Debt:** Credit card companies charge you an enormous amount of interest on each balance that you don't pay off at the end of each month.
- **Credit Card Fraud:** Like cash, sometimes credit cards can be stolen. They may be physically stolen (if you lose your wallet) or someone may steal your credit card number (from a receipt, over the phone, or from a Web site) and use your card to rack up debts.

DEBIT CARD

A debit card (also known as a bank card or check card) is a plastic card that provides an alternative payment method to cash when making purchases. Functionally, it can be called an electronic check, as the funds are withdrawn directly from either the bank account or from the remaining balance on the card. Debit cards may also allow for instant withdrawal of cash, acting as the ATM card for withdrawing cash and as a check guarantee card.

ADVANTAGES OF DEBIT CARDS:

- A consumer who is not credit worthy and may find it difficult or impossible to obtain a credit card can more easily obtain a debit card, allowing him/her to make plastic transactions. For example, legislation often prevents minors from taking out debt, which includes the use of a credit card, but not online debit card transactions.
- For most transactions, a check card can be used to avoid check writing altogether. Check cards debit funds from the user's account on the spot, thereby finalizing the transaction at the time of purchase, and bypassing the requirement to pay a credit card bill at a later date, or to write an insecure check containing the account holder's personal information.
- Like credit cards, debit cards are accepted by merchants with less identification and scrutiny than personal checks, thereby making transactions quicker and less intrusive. Unlike personal checks, merchants generally do not believe that a payment via a debit card may be later dishonored.
- Unlike a credit card, which charges higher fees and interest rates when a cash advance is obtained, a debit card may be used to obtain cash from an ATM or a PIN-based transaction at no extra charge, other than a foreign ATM fee.

DISADVANTAGES OF DEBIT CARDS:

- Use of a debit card is not usually limited to the existing funds in the account to which it is linked, most banks allow a certain threshold over the available bank balance which can cause overdraft fees if the users transaction does not reflect available balance.
- Many banks are now charging over-limit fees or non-sufficient funds fees based upon pre- authorizations, and even attempted but refused transactions by the merchant (some of which may be unknown until later discovery by account holder).
- Many merchants mistakenly believe that amounts owed can be "taken" from a customer's account after a debit card (or number) has been presented, without agreement as to date, payee name, amount and currency, thus causing penalty fees for overdrafts, over-the-limit, amounts not available causing further rejections or overdrafts, and rejected transactions by some banks.
- In some countries debit cards offer lower levels of security protection than credit cards. Theft of the users PIN using skimming devices can be accomplished much easier with a PIN input than with a signature-based credit transaction. However, theft of users' PIN codes using skimming devices can be equally easily accomplished with a debit transaction PIN input, as with a credit transaction PIN input, and theft using a signature-based credit transaction is equally easy as theft using a signature-based debit transaction.
- In many places, laws protect the consumer from fraud much less than with a credit card. While the holder of a credit card is legally responsible for only a minimal amount of a fraudulent transaction made with a credit card, which is often waived by the bank, the consumer may be held liable for hundreds of dollars, or even the entire value of fraudulent debit transactions. The consumer also has a shorter time (usually just two days) to report such fraud to the bank in order to be eligible for such a waiver with a debit card, whereas with a credit card, this time may be up to 60 days. A thief who obtains or clones a debit card along with its PIN may be able to clean out the consumer's bank account and the consumer will have no recourse.

Introduction to Email Security

How Email Works

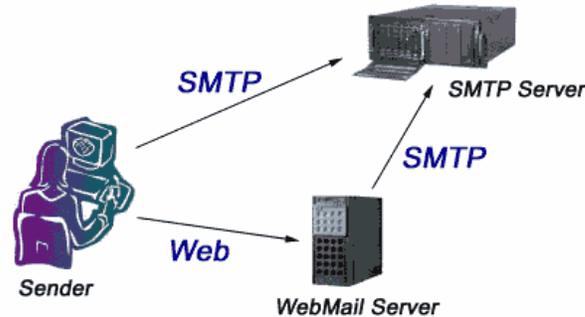
This section describes the general mechanisms and paths taken by an email message on its route from sender to recipient. This should give you an overview of the different protocols (languages) involved, the different types of servers involved, and the distributed nature of email networks. The examples I present are representative of many common email solutions, but are by no means exhaustive.

Sending an Email Message

Sending an email message is like sending a postal letter. When sending a letter, you drop it off at your local post office. The local post office looks at the address and figures out which regional post office the letter should be forwarded to. Then the regional post office looks at the address and figures out which local post office is closest to your recipient. Finally, the recipient's local post office delivers your letter to its recipient. Computers are

UNIT-II: CYBER SECURITY

like “post offices”, and the “Simple Mail Transport Protocol” (SMTP) is the “procedure” which an “email post office” uses to figure out where to send the letter next (e.g. the “next hop”). Any program that sends an email message uses SMTP to deliver that message to the next “post office” for “relaying” it to its final destination.



Most people send mail in two ways – with a web-based interface like Gmail or LuxSci WebMail, or with an “email client” program like Outlook, Thunderbird, iPhone, Android, or Mac Mail

When you send a message with an email program on your personal computer (or your mobile phone or tablet), you have to specify an “SMTP server” so that your email program knows where to send the message. This SMTP server is like your local post office. Your email program talks directly to the server using the computer protocol (language) known as SMTP. This is like dropping off a letter at the local post office.

When you use WebMail, your personal computer uses an Internet connection to communicate with a web server. The “language” that the internet connection uses is HTTP – “HyperText Transfer Protocol”. When you send your message with WebMail, the web server itself takes care of contacting an SMTP server and delivering your message to it.

Delivery of email from your SMTP Server to your recipient’s SMTP Server:

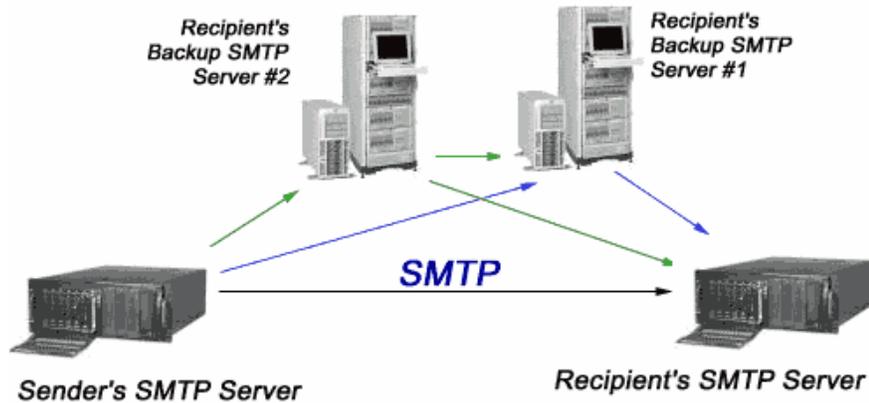
When an SMTP Server receives an email message, it first checks if an account for the message recipient is configured on the server itself. If there is such an account, the server will drop the message in that person’s Inbox (or follow other more complex user defined rules). If there is no local account for that recipient, the server must “relay” the email message to another SMTP server closer to the recipient. This is analogous to how your local post office forwards your letter to a regional post office *unless it is for someone served by the post office itself*. (Post offices don’t actually work this way in general, but the concept is easily understood with this analogy.) This process is known as “SMTP relaying”.

How does your SMTP Server know where to relay the message to?

If the recipient’s email address is “bob@luxsci.net”, then the recipient’s domain name is “luxsci.net”. Part of the “DNS settings” for the recipient’s domain (these are the “mail exchange” or MX records for the domain; see also Understanding Domain Name Service (DNS)) includes an ordered list of SMTP Servers that expect to receive email for this recipient. The highest priority SMTP Server listed (the one with the smallest numerically

UNIT-II: CYBER SECURITY

priority number in the DNS settings) is the recipient's actual inbound SMTP Server; the others are "backup inbound SMTP Servers". These backup servers merely may either queue email for later delivery to the recipient's actual SMTP Server or may perform the same real-time delivery actions as the main SMTP server (e.g. they are there for redundancy)



There are many scenarios that govern the path an email message may take from the sender's to the recipient's SMTP Server. Some of these include:

1. The sender's server successfully contacts the recipient's server and sends the email message directly (black line in the figure).
2. The sender's server cannot contact the recipient's actual SMTP server (maybe the recipient's server is busy, down, or has some other connection problem). In this case the sender's server tries to contact and deliver the message to the recipient's first backup server.
3. The sender's server cannot contact the recipient's actual SMTP server or its first backup server. In this case the sender's server tries to contact and deliver the message the recipient's second backup server.
4. The sender's server cannot contact any of the recipient's servers. In this case it will queue the message and try to send it later. It will keep retrying periodically for several days until it succeeds in sending or gives up.

Any message delivered to the backup servers which queue email goes through the same process of trying to contact the recipient's main SMTP Server, or a higher priority backup servers. Backup servers may also queue email for later sending. (Note that a recipient may have zero or more backup servers, not necessarily two as in this example).

Note: a sender's and recipient's email systems may have arbitrarily complex server configurations actually handling the delivery process ... performing operations such as: backups, filtering, forwarding, queueing, etc. Any number of servers may be involved and and number of copies or backups of messages may be made.

Once the email message arrives at the recipient's SMTP Server and is finally delivered to the recipient's email box, the recipient may pick up the message and read it whenever s/he chooses (as discussed below).

UNIT-II: CYBER SECURITY

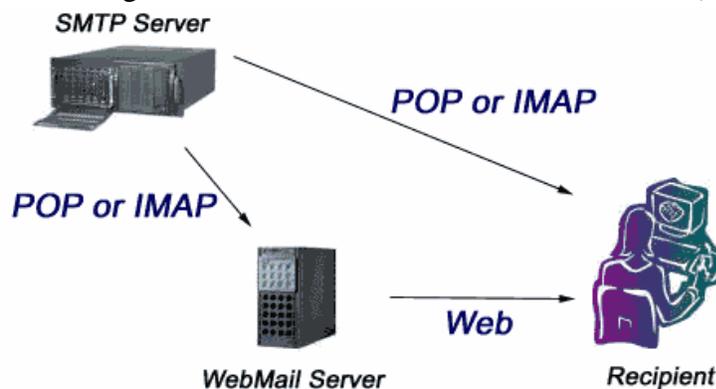
Each server that receives your message adds its own “Received” stamp to the message headers. This stamp identifies what server received the message, at what time, and *from what other server*. This information allows the recipient to trace a message’s entire journey.

What should be clear from this discussion so far is that:

1. Most email servers communicate with each other using SMTP
2. You never know how long it will take an email message to get from sender to recipient because you don’t know how busy the servers are, how much traffic there is on the Internet, what machines are down for maintenance, etc.
3. Your messages may sit in queues on any number of servers for any amount of time. Some of these servers may belong to third parties (i.e. may not be under the purview of either the sender or the recipient, may be an external email filtering or archival organization, etc.).
4. Your recipients can determine the Internet address and name of the computer from which you are sending your messages, even in the case of your email being spoofed by a spammer. (See: tracing the origins of an email address, and hiding it)

Retrieving Email From an SMTP Server

When you receive an email message it sits in a file (or database) in your email server. If you wish to view this email message you must access this content. Any computer wishing to access your email must speak one of the languages the email Server does. With some exceptions (like MS Exchange), there are really only main 2 languages that email servers understand (for email retrieval, as opposed to email sending, for which they use SMTP), one is called the “Internet Message Access Protocol” (IMAP) and one is called the “Post Office Protocol” (POP). (We will not discuss the details of these here, but you may be interested in Understanding Email Services for information about them.)



As a recipient, you can generally retrieve your email by either using a web-based interface known as “WebMail”, or via an “email client” program, such as Microsoft Outlook or iPhone Mail, running on your personal computer or device. The email client programs will talk directly to your email server and speak IMAP or POP or something similar. With WebMail, your computer will talk to a WebMail server using a web connection (speaking

HTTP); the WebMail server will, in turn, talk to your email server using POP or IMAP or something similar (like a direct database connection).

The Lack of Security in Email

Email is *inherently insecure*. In the following sections, we will see just how insecure it is. At this stage, it is important to point out the insecurity in the email delivery pathway just discussed:

- **WebMail:** If the connection to your WebMail server is “insecure” (i.e. the address is http:// and NOT https://), then all information including your username and password is not encrypted as it passes between the WebMail server and your computer.
- **SMTP:** SMTP does not encrypt messages (unless the servers in question support opportunistic TLS encryption). Communications between SMTP servers may send your messages in plain text for any eavesdropper to see. Additionally, if your email server requests that you send your username and password to “login” to the SMTP server in order to relay messages to other servers, then these are also sent in plain text, subject to eavesdropping. Finally, messages sent via SMTP include information about which computer they were sent from and what email program was used. This information, available to all recipients, may be a privacy concern.
- **POP and IMAP:** The POP and IMAP protocols require that you send your username and password to login; these credentials are not encrypted. So, your messages and credentials can be read by any eavesdropper listening to the flow of information between your personal computer and your email service provider’s computer.
- **BACKUPS:** Email messages are generally stored on SMTP servers in plain, unencrypted text. Backups of the data on these servers may be made at any time and administrators can read any of the data on these machines. The email messages you send may be saved unexpectedly and indefinitely and may be read by unknown persons as a result.

These are just a few of the security problems inherent in email. In the next section, we will talk about communications security problems in general so we can see what else can go wrong. Later on, we will see *how these problems can be solved*.

Security Threats to Your Email Communications

Eavesdropping: The Internet is a big place with a lot of people on it. It is very easy for someone who has access to the computers or networks through which your information is traveling to capture this information and read it. Just like someone in the next room listening in on your phone conversation, people using computers “near” the path your email takes through the Internet can potentially read and copy your messages!

Identity Theft: If someone can obtain the username and password that you use to access your email servers, they can read your email and send false email messages as you. Very often, these credentials can be obtained by eavesdropping on SMTP, POP, IMAP, or

UNIT-II: CYBER SECURITY

Webmail connections, by reading email messages in which you include this information, or through other means.

Invasion of Privacy: If you are very concerned about your privacy, then you should consider the possibility of “unprotected backups”, listed below. You may also be concerned about letting your recipients know the IP address of your computer. This information may be used to tell in what city you are located or even to find out what your address is in some cases! This is not usually an issue with WebMail, POP, or IMAP, but is an issue with the transport of email, securely or insecurely, from any email client over SMTP.

Message Modification: Anyone who has system administrator permission on any of the SMTP Servers that your message visits, can not only read your message, but they can delete or change the message before it continues on to its destination. Your recipient has *no way to tell* if the email message that you sent has been altered! If the message was merely deleted they wouldn't even know it had been sent.

False Messages: It is very easy to construct messages that appear to be sent by someone else. Many viruses take advantage of this situation to propagate themselves. In general, there is no way to be sure that the apparent sender of a message is the true sender – the sender's name could have been easily fabricated.

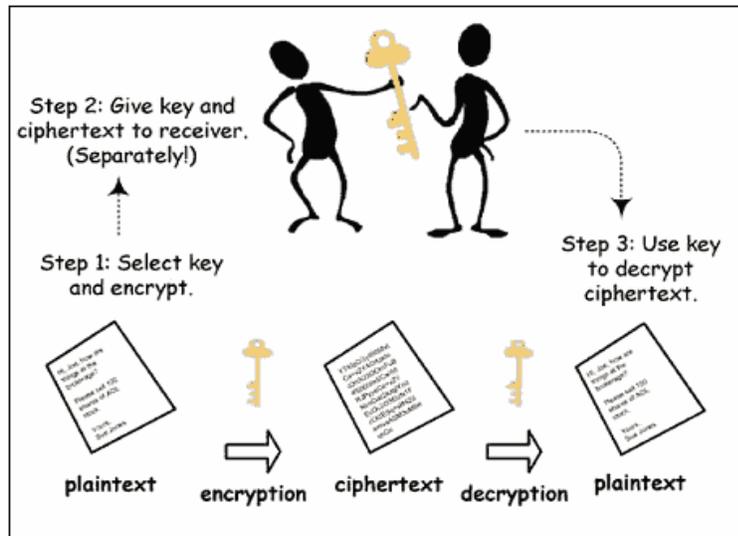
Message Replay: Just as a message can be modified, messages can be saved, modified, and re-sent later! You could receive a valid original message, but then receive subsequent faked messages that appear to be valid.

Unprotected Backups: Messages are usually stored in plain text on SMTP Servers. Thus, backups of these servers' disks usually contains plain text copies of your messages. As backups may be kept for years and can be read by anyone with access to them, your messages could still be exposed in insecure places even after you think that all copies have been “deleted”.

Repudiation: Because normal email messages can be forged, there is no way for you to prove that someone sent you a particular message. This means that even if someone DID send you a message, they can successfully deny it. This has implications with regards to using email for contracts, business communications, electronic commerce, etc.

Symmetric and Asymmetric Encryption in a Nutshell

A basic knowledge of the two main types of encryption will be very useful. This section presents these concepts in a simple, straightforward form.



Symmetric Encryption

In symmetric encryption, you and your friend share a “secret” key. Using this key, you can encrypt a message into “cypher text”. Cypher text looks like a random sequence of characters and is completely meaningless to anyone unless they also have the secret key, in which case they can decrypt the cypher text back into the original message and read it. Using symmetric key encryption, eavesdropping and unwanted backups of your messages no longer are a problem (unless the eavesdropper knows what your secret key is). It also becomes harder for someone to modify your messages in transit in any kind of a meaningful way.

The problem with symmetric key encryption is precisely the fact that you and your friend must share the same secret key. Unless you meet in person, *how do you communicate this key in a way that is secure?* What if you want to send a secure message to someone on the other side of the world? How do you get them the secret key quickly in a way that eavesdroppers can’t detect?

Message Digests / Authentication Codes

A “Message Digest” or “Message Authentication Code” is really a very simple concept. You take your message and pass it through an algorithm that spits out a relatively short sequence of characters (maybe 128 or 256 or so of them). This sequence of characters is a “fingerprint” of the message. Any minute change in the message would produce a significantly different “fingerprint”. There is no way to “reverse engineer” the original message from its fingerprint and it is almost impossible (assuming that your method of making these fingerprints is sufficiently “good”) to find two messages that yield the same fingerprint (just like trying to find two complete strangers who have the same fingerprint). Message Digests are quick ways to check to see if a message has been altered. If you have a digest of the original message and compare it with a digest of the message you just received and they match, then you know that the message has been unaltered.

Asymmetric Encryption

In asymmetric encryption, also known as “public key” encryption, each person has TWO keys. Any cyphertext created using one of the keys can ONLY be decrypted using the other key. For example, say you have keys “K1” and “K2”. If you encrypt your message with K1, then ONLY K2 can be used to decrypt it. Similarly, if you encrypt using K2, ONLY K1 can be used to decrypt it. This is distinctly different from symmetric key encryption where you only have one key that performs both functions on the same message.

In asymmetric key encryption, the two keys that each person possesses are commonly named the “private” and “public” keys because the “public” one is published or given out freely to anyone who wants a copy and the “private” one is kept secret. The security of asymmetric key encryption depends only on whether you can keep your private key secret.

Asymmetric key encryption allows you to do many clever things:

- **Send an Encrypted Message:** To send a secure message to someone, all you have to do is encrypt it with their public key! Only the intended recipient who has the matching private key will be able to decrypt and read the message. This solves the problem of eavesdropping and the problem of sending secret keys that is inherent in symmetric key encryption.
- **Prove You Sent A Message:** To prove to someone that you sent a message, you can encrypt the message (or just a piece of it) with your private key. Then, anyone can decrypt it with your public key and read the contents. The fact that your public key decrypts the message proves that only you could have sent it (or someone who has your private key).
- **Sign a Message:** A message signature proves that you sent the message AND allows the recipient to determine if the message was altered in transit. This is done by using your private key to encrypt a digest of a message at the time of sending. The recipient can decrypt this digest and compare it to a digest of the received message. If they match, then the message is unaltered and was sent by you.
- **Encrypted, Signed Messages:** The most secure form of communication is to first add a signature to the message and then to encrypt the message plus signature with the recipient’s public key. This combines all of the benefits of all of the techniques: security against eavesdropping and unexpected storage, proof of sender, and proof on message integrity.

Securing Your Email With SSL or TLS

The easiest thing you can do to make your email more secure is to use an email provider that supports “Secure Socket Layer” (SSL) for their Webmail, POP, IMAP, and SMTP servers. TLS (in common parlance) is a type of SSL that can be initiated during a mail session; unlike TLS, SSL must be initiated before sending the email.

SSL is a combination of asymmetric and symmetric key encryption mechanisms. If you connect to a server using SSL, the following things happen (roughly):

UNIT-II: CYBER SECURITY

1. The server uses its private key to prove to you that it is in fact the server that you are trying to connect to. This lets you know that you are not connecting to a “middleman” that is trying to intercept your communications.
2. You send the server your public key.
3. The server generates a “secret key” and sends it to you encrypted using your public key.
4. You and the server then communicate using symmetric key encryption using this shared secret key. (Symmetric key encryption is faster than asymmetric key encryption).

The benefits of SSL are twofold: 1. you can determine if you are connecting to the right server, and 2. you and the server can communicate securely.

If you get any warning messages when connecting to a server using SSL, you should *think twice about ignoring them*. While your provider may just have a small technical problem that is causing the warning, these warnings can also indicate that your communications *are being intercepted*. These warnings usually indicate one of the following:

1. The server’s SSL “certificate” (i.e. public/private key pair) has expired.
2. Some of the information in the certificate doesn’t match the information you expect — i.e. the certificate was issued for a different server name than the one you are trying to connect to. (You could be inadvertently connecting to the wrong server.)
3. The certificate was issued by an untrusted agency.

SSL certificates are (generally) issued by third party agencies such as Thawte.com or VeriSign. These 3rd party companies do a background check on companies that request certificates, and only issue certificates if the companies have a right to them. The certificate includes the name of the company, the name of the issuing company, and the name of the server to which it is issued. When you connect to an SSL server you can verify this embedded information and the fact that it was issued by a third party company that you trust. If the certificate is valid then you can have a high degree of confidence that the server you are connecting to is the server you want to reach.

By using SSL for WebMail, POP, IMAP, and SMTP, you ensure that communications between your personal computer and your email service provider’s servers will be encrypted. Your message contents, username, and password will be hidden from eavesdroppers — *but only hidden from eavesdroppers between you and your service provider!* SSL services do not protect your messages once they arrive to your SMTP Server or when they leave your SMTP Server and head to their destinations. So, it *doesn’t really protect your message contents*, but it does completely protect your username and password from detection. This is very important because it prevents identity theft, forged messages, etc.

SSL is very easy to use. It usually only involves clicking a few checkboxes in the configuration of your email client. It is transparent to your recipients – you can use SSL for these services even if your recipients do not. These measures protect you and your

password. Because it is so easy and because the security you receive is much better than no security, we strongly encourage the use of SSL for email communications whenever possible.

Privacy with Anonymous SMTP

In Sections 2 and 3 we wrote that when you send email via any email client (not usually WebMail ... though some provider's WebMail also does this), your computer's Internet IP address is included in the message *for all recipients to see*. Depending your Internet Service Provider's privacy standards and what kind of connection and service you have, this information may be used to determine what region or the country you are in, what city you are in, or even what your address is! This could be a serious issue for people very concerned about their privacy. Additionally, other information such as what email program you are using is also visible to the recipient.

Anonymous SMTP services, or re-mailers, provide a good way of keeping all of the functionality of SMTP that you require, while giving you back your privacy. These services typically receive your message via (Authenticated) SMTP (ideally they would also support SMTP over SSL, as described above) and then “**scrub**” the message, removing all information about your computer's address, your email program, and any other non-standard information. These services then re-email your scrubbed message to the intended recipients.

The end result is that the recipients get the message just as they would have without the “Anonymous” service, except that they can only track the message back to your Anonymous SMTP server. They know who you are, based on your email address and your message content, but they have no way of knowing **where you are** or what email programs you are using.

Most anonymous SMTP services log all information that they scrub out of messages and track all activity. So, while your recipients do not know where you are, your email service provider *does*. This prevents this type of service from providing any real benefit to people who send unsolicited or forged email — their service provider can quickly respond to complaints or abuse, identify the sender, and terminate the account and/or bring legal measures to bear.

What it does provide is a level of privacy in sending of email that is functionally equivalent to the level of privacy you get from sending through a WebMail interface (unless the WebMail in question makes it a point to add your computer's address into the outgoing message — most do not).

Asymmetric Key Encryption and Email (PGP and S/MIME)

While SSL protects your password and your message contents to some extent, it does not solve any of the other problems we have discussed: repudiation, encryption, unwanted backups, message modification, etc. This is because SSL only protects the message path between you and your SMTP Server *and stops there*. Even with SSL, the messages are stored on your SMTP Server in *plain text*.

UNIT-II: CYBER SECURITY

The ultimate solution is to use asymmetric key encryption to provide message signatures and/or encryption. This completely solves the issues of:

- Eavesdropping (everything is always encrypted)
- Message modification (message digests are used)
- Message replay (you can include a timestamp in the signature)
- Repudiation (signatures allow proof of who sent the message)
- Unprotected backups (everything is always encrypted)

Asymmetric key encryption should be used in combination with SSL so that your username and password are also protected. Why? *These credentials are not part of the message and thus would not be encrypted along with the message.*

Fortunately (or unfortunately), there are two widely used forms of asymmetric key encryption for email: S/MIME and PGP. Both allow you to add signatures and/or encryption to your messages. PGP can be obtained from PGP.com and is compatible with standard email clients. S/MIME is built into many email clients like Microsoft Outlook, but you must obtain an S/MIME certificate from a third-party company such as Thawte.com.

Interoperability Problems

PGP and S/MIME solve many problems, but they also create another: interoperability. One interoperability issue is that PGP and S/MIME are completely incompatible! If you are using PGP and your friend is using S/MIME, you will not be able to send each other secure messages.

That said, PGP has been an Internet standard (OpenPGP – RFC 2440) since 1997 and PGP-encrypted email accounts for well over 90% of the current encrypted email traffic on the Internet. So, using PGP will make you compatible with the majority. However, the majority doesn't matter if you're trying to contact the minority that use S/MIME. It is useful to know that some email clients, such as Microsoft Outlook, can be configured to use BOTH PGP and S/MIME so that you can correspond securely using whatever method is necessary at the moment. Additionally, most major email programs have support for S/MIME built in (and not PGP) ... so it is much easier and cheaper to get started with S/MIME.

The other interoperability issue involves “key exchange”. If you want to send your friend an encrypted message, you first need his/her public key; if your friend wants to prove that you signed a message or that the message that you sent him/her was unaltered, s/he first needs your public key. So there is the necessity of trading public keys before secure communication can begin. There are various ways to trade keys (including email). PGP offers “key servers” from which your correspondents' keys can be downloaded to make the process easier. However, not everyone has their PGP keys listed on a key server, let alone the same key server, and not everyone uses PGP, so the key exchange issue is still an impediment to sending secure messages — especially if you have to send them quickly.

Compatible Security with “Secure Message Pickup”

Secure Message Pickup Encryption uses a trusted “encryption middleman” to give you the almost same level of security offered by asymmetric key encryption, but with universal compatibility. Here is how it works:

1. The sender connects to the middleman’s WebMail portal on a secure SSL connection
2. The middleman validates the sender.
3. The sender creates a message.
4. The message sender chooses some method for the recipient’s identity to be verified (e.g. via a password, a question an answer, a login to a portal, etc.)
5. The middleman encrypts the message (e.g. using PGP or AES256) and stores it on his server.
6. The middleman sends a plain text message to the recipient that contains only a secure link to the middleman’s web mail portal, and a unique message password that is part of the encryption key. The middleman then ‘forgets’ this password so that he **cannot** decrypt the message until he gets the password back from the recipient.
7. The recipient connects to the middleman’s portal over a secure SSL connection and logs in (the message password coming along for the ride).
8. The middleman decrypts the message and presents it to the recipient.

The encryption middleman handles all the encryption dirty work; it doesn’t matter if the sender uses PGP and the recipient uses S/MIME. In fact, it doesn’t matter if either uses encryption at all! All that the sender and recipient need is a web browser and regular email service. The middleman takes care of everything else.

How does it solve the security problems we mentioned earlier?

1. Eavesdropping: No one can eavesdrop on the message because the sender and recipient connect to the middleman on a secure SSL connection.
2. Identity Theft: No one can steal the sender’s login information or the recipient’s verification information because both the sender and the recipient use SSL connections.
3. Invasion of Privacy: The recipient knows nothing about the sender’s computer, email client, or location. She only knows that he used the middleman.
4. Message Modification: No one can modify the message because it never leaves the middleman’s server and is encrypted and signed while residing there.
5. False Messages: The message is only accessed on the middleman’s server, so no one else can pretend to send it.
6. Message Replay: No one can re-send the message because it never leaves the middleman’s server.
7. Unprotected Backups: The message is encrypted when it is stored, so it is secure even in backups.

UNIT-II: CYBER SECURITY

8. Repudiation: The recipient knows that the sender really did send the message because he was validated by the middleman and because digital signatures are used.

In addition the middleman can keep a log of who accesses the message and at what times. Thus the sender can audit the message to see who has viewed it. Notice that the message is secure and anonymous; The message is encrypted and stored on the middleman's servers, so it is not subject to the security of intermediate relaying servers. Only the middleman can encrypt and decrypt the message, and only authorized recipients can access the message. The recipient knows nothing about the sender's computer, only that he used the middleman. *As long as the middleman is trustworthy*, the message is completely secure, completely anonymous, and completely compatible.

SMTP TLS

There is a growing trend for SMTP Servers to support "opportunistic TLS." If the sending server and the recipient server both support opportunistic TLS, then they will automatically negotiate a secure TLS-encrypted connection and transport your email message in that manner. This has the effect of securing server-to-server delivery of email across the Internet. This is great, except:

1. A majority of email providers do not yet support Opportunistic TLS
2. If it is not supported or security setup fails, your email will be delivered in the old insecure manner.

An alternative to opportunistic TLS that is supported by LuxSci as part of SecureLine is "Forced TLS". With Forced TLS,

1. Email that is going to a recipient whose server support opportunistic TLS will be delivered via TLS *or not delivered at all until TLS is working*.
2. Email that is going to a recipient whose server does NOT support opportunistic TLS will "fall back" to use of some other form of encryption (e.g. Secure Message Pickup)

When TLS use "good enough" for you it is ideal, as it is easiest form of encryption to use, being transparent to the sender and recipient. It is "business as usual", which everyone likes. So, if TLS is good enough, use TLS with recipients who support it and use other means with those that do not.

Conclusion

Email is, in general, *Completely Insecure!* The security issues include:

1. Eavesdropping
2. Identity Theft
3. Invasion of Privacy
4. Message Modification
5. False Messages
6. Message Replay
7. Unprotected Backups
8. Repudiation (Sender denies that s/he sent it)

UNIT-II: CYBER SECURITY

SSL: It is simple and easy to use SSL (and TLS) to secure the communications between your computers and your email service provider's computers. This works no matter who your recipients are. SSL improves security in these ways:

1. It establishes that you are contacting your service provider's computers and not someone else's
2. It encrypts the username and password that you use to login to these servers. This mitigates identity theft and other issues.
3. It protects your message from eavesdroppers between your computer and your SMTP server.

Anonymity: If you have access to an Anonymous SMTP server, you have an easy way to increase your Internet privacy. Anonymous SMTP provides:

1. IP address privacy so that message recipients cannot determine your computer's Internet address (and thus your location).
2. Email client privacy so that the recipients of your email messages cannot determine what type of email client you are using.
3. A means to strip out any other non-standard "email header" data that may be lurking in your outbound messages.

PGP and S/MIME: PGP and S/MIME keys use asymmetric key encryption to protect the contents of your messages throughout their complete journeys. They provide:

1. Protection against eavesdropping and unwanted backups
2. Message Digests to detect whether messages have been altered in transit
3. Signatures to prove sender authenticity.