UNITIINTRODUCTION

Security trends - Legal, Ethical and Professional Aspects ofSecurity, Need for Security atMultiple Levels, Security Policies - Model of Network Security - Security Attacks, Services andMechanisms-OSIsecurityarchitecture-classicalencryptiontechniques:substitutiontechniques, transposition techniques, steganography - Foundations of modern cryptography:perfectsecurity-informationtheory-productcryptosystem-cryptanaysis

1.1 SECURITYTRENDS

Theprotectionaffordedtoanautomatedinformationsysteminordertoattaintheapplicable objectives of preserving the integrity, availability, and confidentiality of informationsystemresources(includeshardware,software,firmware,information/data,andtelecom munications)

Thisdefinitionintroduces threekeyobjectives that areat theheartofcomputer security:

- **Confidentiality:** Thistermcoverstworelatedconcepts:
- **Dataconfidentiality:**Assuresthatprivateorconfidentialinformationisnotmadeavailableordis closed tounauthorizedindividuals.
- **Privacy:** Assures that individuals control or influence what information related to themmaybecollectedandstoredandbywhomandtowhomthatinformationmaybedisclosed.
- **Integrity:**Thisterm covers tworelatedconcepts:
- **Data integrity:** Assures that information and programs are changed only in a specifiedandauthorizedmanner.
- **Systemintegrity:**Assuresthatasystemperformsitsintendedfunctioninanunimpaired manner, free from deliberate or inadvertent unauthorized manipulation of thesystem.
- **Availability:**Assuresthatsystemsworkpromptlyandserviceisnotdeniedtoauthorizedusers

These three concepts form what is often referred to as the **CIA triad** (Figure 1.1). The threeconcepts embody the fundamental security objectives for both data and for information andcomputingservices

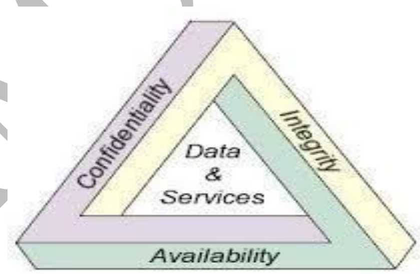


Figure1.1CIAtriad

Although the use of the CIA triad to define security objectives is well established, some in thesecurity field feel that additional concepts are needed to present a complete picture. Two of themostcommonlymentionedareasfollows:

- **Authenticity:**Thepropertyofbeinggenuineandbeingabletobeverifiedandtrusted;confidence in the validity of a transmission, a message, or message originator. This meansverifying that users are who they say they are and that each input arriving at the system camefroma trustedsource.

• **Accountability:** The security goal that generates the requirement for actions of an entity to betraced uniquely to that entity. This supports non repudiation, deterrence, fault isolation, intrusiondetectionandprevention,and after-action recovery andlegalaction.

• **Computer Security -** Generic name for the collection of tools designed to protect data and tothwarthackers**.**

• **NetworkSecurity-**Measurestoprotect dataduring theirtransmission.

• **Internet Security** - Measures to protect data during their transmission over a collection ofinterconnected networks Our Focus is on Internet Security which consists of measures to deter,prevent,detect and correct security violations thatinvolve the transmission and storage ofinformation

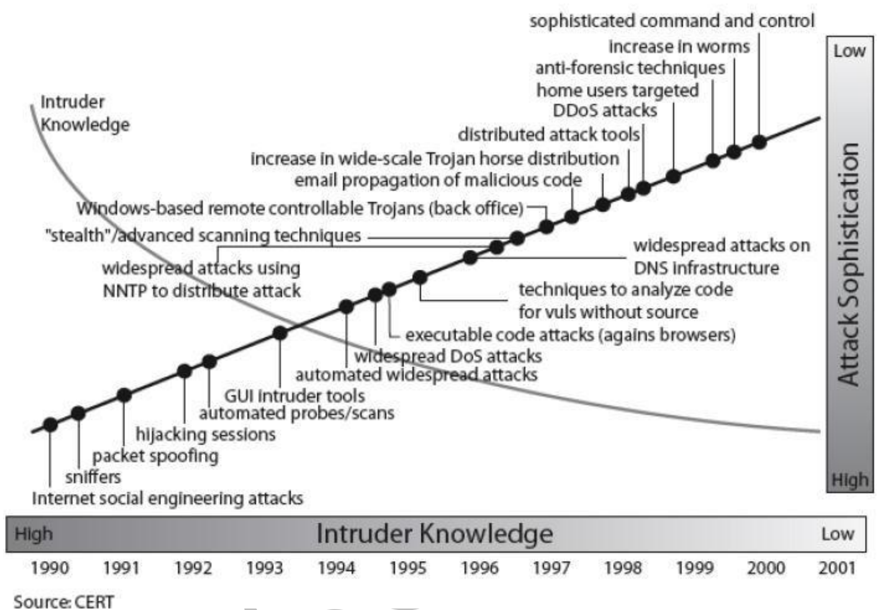


Figure1.2SecurityTrends

1.1.1THECHALLENGESOFCOMPUTERSECURITY

Computerandnetworksecurityisbothfascinating andcomplex.Someof thereasonsfollow:

1. Security is not as simple as it might first appear to the novice. The requirements seem to bestraightforward; indeed, most of the major requirements for security services can be given self-explanatory,one-wordlabels:confidentiality,authentication, nonrepudiation,orintegrity

2. Indevelopingaparticularsecuritymechanismoralgorithm,onemustalwaysconsiderpotentialattac kson thosesecurityfeatures.

3. Typically, a security mechanism is complex, and it is not obvious from the statement of aparticularrequirementthatsuch elaboratemeasuresareneeded.

4. Having designed various security mechanisms, it is necessary to decide where to use them.Thisis truebothin terms ofphysicalplacementandinalogicalsense

.

5. Securitymechanismstypicallyinvolve morethanaparticularalgorithm orprotocol

6. Computer and network security is essentially a battle of wits between a perpetrator who triestofindholesandthedesigneroradministratorwhotriestoclosethem.Thegreatadvantagethat the attacker has is that he or she need only find a single weakness, while the designer mustfindandeliminate allweaknessestoachieveperfectsecurity.

7. There is a naturaltendencyonthe partofusersandsystem managerstoperceive littlebenefitfromsecurity investmentuntilasecurityfailure occurs.

8. Securityrequiresregular,evenconstant,monitoring,andthisisdifficultintoday"sshort-term,overloadedenvironment.

9. Securityisstilltoooftenanafterthoughttobeincorporatedintoasystemafterthedesigniscomplete ratherthan beinganintegralpartofthedesignprocess.

10. Manyusersandevensecurityadministratorsviewstrongsecurityasanimpedimenttoefficient anduser-friendly operationofaninformationsystem or useofinformation.

1.2 LEGAL,ETHICALANDPROFESSIONAL ASPECTSOFSECURITY

Today millions of people perform online transactions every day. There many ways to attackcomputer and networks to take advantage of what has made shopping, banking, transformationofmessages, investments and leisure pursuits a simple matter ofdraggingand clicking formany people. Thus, the laws and ethics are important aspects in data and network security.The legal system has adapted quite well to computer technology by reusing some old forms oflegal protection (copyrights and patents) and creating laws where no adequate one existed(malicious access).Still the courts are not a perfect form of protection for computer, for tworeasons, firstcourt tends to be reactive instead ofproactive. Thatis,we have towaitforregression to occur and then adjudicative it, rather than try to prevent it in first place. Secondfixingaproblemthroughthecourts canbe timeconsumingandmoreexpensive.

The latter characteristic prevents all but the wealthy from addressing most wealthy. Onother hand, 1ethics has not had to change , because ethic is more situational and personal thanthe law, for example the privacy of personal information becoming important part of computernetworksecurityandalthoughtechnicallythisissueisjustanaspectofconfidentiality,practically ithasalonghistoryinboth law andethics.

Law and security are related in several ways. First international, national, state, city lawsaffect privacy, secrecy. These statutes often apply to the rights of individuals to keep personalmatters private. Second law regulates the use of development, and ownership of data andprograms.Patents,copy rights,and tradesecretsarelegal devicestoprotecttheright ofdevelopersandownersoftheinformationand data.

1.2.1 Cryptographyand Law

Cyber-Crime: - Criminal activities or attacks in which computer and computer networks are tool,target, or place of criminal activity. Cybercrime categorize based on computer roles such astarget,storagedeviceandcommunication tool.

Computersastargets: Togettheinformationfromthecomputersystemorcontrolthecomputer system without the authorization or payment or alter the interfaces or data in theparticularsystemwithuseofserver.

Computers as storage devices: Computers can be used to further unlawful activity by using acomputeroracomputerdeviceasapassivestoragemedium.Forexample,thecomputercanbe usedtostore stolenpasswordlists, credit carddetailsandproprietarycorporate information.

Computers as communications tools: Many of the crimes falling within this category aresimplytraditionalcrimesthatarecommittedonline.Examplesincludetheillegalsaleofprescriptiondrugs,controlledsubstances,alcohol,andguns;fraud;gambling;andchildpornography. Other than these crimes there are more specific crimes in computer networks.Thereare:

Illegal access: Theaccesstothewholeoranypartofacomputersystemwithoutright.Illegalinterception:Theinterceptionwithoutright,madebytechnicalmeans,ofnon-publictransmissions of computer data to, from or within a computer system, including electromagneticemissions froma computersystemcarryingsuchcomputerdata.

Data interference: The damaging, deletion, deterioration, alteration or suppression of computerdatawithoutright.

System interference: The serious hindering without right of the functioning of a computersystembyinputting,transmitting,damaging,deleting,deteriorating,alteringorsuppressingcomputerdata.

Computer-related forgery: The input, alteration, deletion, or suppression of computer data,resulting in inauthentic data with the intent that it be considered or acted upon for legal purposesasifitwere authentic, regardlesswhetherornot the datais directlyreadable andintelligible.

Crime related to child pornography: Producing child pornography or distribution through acomputer system and making available or distributing or transmitting child pornography throughacomputersystem.

The relative lack of success in bringing cyber-criminals to justice has led to an increase in theirnumbers, boldness, and the global scale of their operations. It is difficult to profile cybercriminalsin the way that is often done with other types of repeat offenders. The success of cybercriminalsand the relative lack of success of law enforcement, influence the behaviour of cybercrimevictims. As with law enforcement, many organizations that may be the target of attack have notinvestedsufficientlyin technical,physical,andhuman-factor resourcesto preventattacks.

Thelawisusedregulatepeoplefortheirowngoodandforthegreatergoodofsociety.Cryptography alsoregulatedactivity.

SomeExample lawswhichareforcedon cryptography.

Control use of cryptography: Closely related to restrictions on content are restrictions on theuse of cryptography imposed on users in certain countries. For examples, 2 In China, statecouncilorder273requiresforeignorganizationsorindividualstoapplypermissiontouseencryption in China. Pakistan requires that all encryption hardware and software be inspectedandapprovedby the Pakistantelecommunicationauthority.

Cryptography and Free speech: The Cryptography involve not just products, it involves ideastoo, although governments effectively control the flow of products across borders, controlling thefloeideaseitherheadorontheinternet,isalso impossible.

CryptographyandEscrow: Althoughlawsenablegovernmentstoreadencryptedcommunications.
In 1996, US government offered to relax the export restriction for so calledescrowed encryption, in which the government would able to obtain the encryption key for anyencryptedcommunication.

The victory in use of law enforcement depends much more on technical skills of the people.Managementneedstounderstandthecriminalinvestigationprocess,theinputsthatinvestigatorsneed,andthewaysinwhichthevictimcancontributepositivelytotheinvestigation.

1.2.2 IntellectualProperties.

Therearethreemaintypesofintellectualpropertyforwhichlegalprotectionisavailable. **Copy rights:** Copyright law protects the tangible or fixed expression of an idea, not the ideaitself. Copy right properties exists when proposed work is original and creator has put originalidea in concrete form and the copyright owner has these exclusive rights, protected againstinfringementsuchasreproduction right,modificationright,distributionright

Patents: A patent for an invention is the grant of a property right to the inventor. There are 3typesinpatents:-

- Utility(anynewandusefulprocess,machine,articleofmanufacture,orcompositionofmatter).
- Design(new,original,andornamentaldesignforanarticleofmanufacture)
- Plant(discoversandasexuallyreproducesanydistinctand newvarietyofplant).

Trade-Marks: A trademark is a word, name, symbol or expression which used to identify theproducts or services in trade uniquely from others. Trade mark rights used to preventothersfrom using a confusingly similar mark, but not to prevent others from making the same goods orfromsellingthesamegoodsor servicesunderaclearlydifferentmark.

- Intellectual Property Relevant to Network and Computer SecurityA number of forms of intellectual property are relevant in the context of network andcomputersecurity.
- Softwareprograms:softwareprogramsareprotectedbyusingcopyright,perhapspatent.
- Digitalcontent:audio/video/ media / web protected by copy rightAlgorithms:algorithmsmay beableto protectbypatenting
- PrivacyLawandRegulation:Anissuewithconsiderableoverlapwithcomputersecurityis that ofprivacy. Concerns about the extent to which personal privacy has been andmaybecompromisedhaveledtoavarietyoflegalandtechnicalapproachestoreinforcingprivacyrights.Anumberofinternationalorganizationsandnationalgovernmentshaveintroducedlawsandregulationsintendedtoprotectindividualprivacy.
- EuropeanUnionDataProtectionDirectivewasadoptedin1998 to ensurememberstates protect fundamental privacy rights when processing personal info and preventmember states from restricting the free flow of personal info within EU organized aroundprinciplesofnotice,consent,consistency,access,security,onwardtransferandenforcement. US Privacy Law have Privacy Act of 1974 which permits individuals todetermine records kept, forbid records being used for other purposes, obtain access torecords,ensuresagenciesproperlycollect,maintain,andusepersonalinfoandcreatesa private right of action for individuals.Cryptography andEthics.

- Therearemanypotentialmisusesandabusesofinformationandelectroniccommunication that create privacy and security problems. Ethics refers to a system ofmoral principles that relates to the benefits and harms of particular actions. An ethic anobjectively defined standard of right and wrong. Ethical standards are often idealisticprinciplesbecausetheyfocusononeobjective.Eventhoughreligiousgroupandprofes sional organization promote certain standards of ethical behaviour, ultimately eachpersonis responsiblefor decidingwhatdoinaspecific situation.

1.2.3 Ethicalissuesrelatedtocomputerandinfo systems

Computers have become the primary repository of both personal information and negotiableassets,suchas bankrecords,securities records,andotherfinancialinformation.

Repositoriesandprocessorsofinformation: Unauthorizeduseofotherwiseunusedcomputer services or of information stored in computers raises questions of appropriateness orfairness.

Producers of new forms and types of assets: For example, computer programs are entirelynewtypesofassets,possibly notsubjecttothesameconcepts ofownershipasotherassets.

Symbols of intimidation and deception: The images of computers as thinking machines,absolute truth producers, infallible, subject to blame, and as anthropomorphic replacements ofhumanswhoerrshouldbecarefullyconsidered.

1.3 NEEDFORSECURITYATMULTIPLELEVELS

Multilevel security or multiple levels of security (MLS) is the application of a computer system toprocess information with incompatible classifications (i.e., at different security levels), permitaccess by users with different security clearances and needs-to-know, and prevent users fromobtainingaccess toinformationforwhichtheylackauthorization.

Therearetwo contextsfor the useof multilevel security.

One is to refer to a system that is adequate to protect itself from subversion and has robustmechanismstoseparateinformationdomains,thatis,trustworthy.

Another context is to refer to an application of a computer that will require the computer to bestrong enough to protect itself from subversion and possess adequate mechanisms to separateinformation domains, that is, a system we must trust. This distinction is important becausesystemsthatneed to be trustedarenotnecessarilytrustworthy.

A threatis anobject,person,orotherentity thatrepresentsa constantdanger toan asset.

1.3.1 SecurityPolicies

The CryptographyPolicy setsoutwhenandhow encryption shouldbeused.Itincludesprotection of sensitive information and communications, key management, and procedures toensureencryptedinformationcanberecoveredbytheorganisationifnecessary.

Role oftheSecurityPolicyin SettingupProtocols

Following are some pointers which help in setting u protocols for the security policy of anorganization.

- Who shouldhaveaccess to the system?
- Howitshould beconfigured?
- Howto communicatewiththirdparties orsystems?

Policiesaredividedin twocategories:

- Userpolicies
- IT policies.

Userpoliciesgenerallydefinethelimitoftheuserstowardsthecomputerresourcesinaworkplace.Forexample, whataretheyallowedto install in theircomputer,iftheycanuseremovablestorages?

Whereas, IT policies are designed for IT department, to secure the procedures and functions ofITfields.

- **General Policies** − This is the policy which defines the rights of the staff and access leveltothesystems.Generally,itisincludedevenin thecommunicationprotocolasapreventivemeasurein casethereareanydisasters.
- **ServerPolicies**−Thisdefineswhoshould haveaccesstothespecificserverand withwhat rights. Which software's should be installed, level of access to internet, how theyshouldbeupdated?
- **Firewall Access and Configuration Policies** − It defines who should have access to thefirewall and what type of access, like monitoring, rules change. Which ports and servicesshouldbeallowedandifitshouldbe inboundoroutbound?
- **BackupPolicies**−Itdefineswhoisthe responsibleperson forbackup,whatshouldbethebackup,whereitshouldbebackedup,howlongitshouldbekeptand thefrequencyofthebackup.
- **VPN Policies** − These policies generally go with the firewall policy; it defines those userswhoshouldhaveaVPN accessandwithwhatrights.Forsite-to-site connectionswithpartners, itdefinestheaccess levelofthepartner toyournetwork,typeof encryption tobeset.

1.3.2 StructureofaSecurityPolicy

When youcompileasecuritypolicy youshould haveinmindabasicstructure in order to makesomething practical.Some ofthemainpointswhich haveto be takenintoconsiderationare:

- DescriptionofthePolicy andwhatistheusagefor?
- Wherethispolicyshouldbeapplied?
- Functionsandresponsibilitiesoftheemployees that areaffectedbythispolicy.
- Proceduresthatareinvolved inthis policy.
- Consequencesifthepolicyis notcompatiblewith companystandards.

TypesofPolicies

- **Permissive Policy** − It is a medium restriction policy where we as an administrator blockjustsome well-known ports ofmalware regardinginternetaccess and justsome exploitsare takeninconsideration.
- **Prudent Policy** − This is a high restriction policy where everything is blocked regarding theinternet access, just a small list of websites is allowed, and now extra services are allowedincomputerstobeinstalledandlogsaremaintainedforevery user.

- **Acceptance User Policy** − This policy regulates the behavior of the users towards asystem or networkor even awebpage, so it is explicitly saidwhat a user can do andcannotinasystem.Likearetheyallowedtoshareaccesscodes,cantheyshareresources,etc.
- **User Account Policy** − This policy defines what a user should do in order to have ormaintainanotheruserinaspecificsystem.Forexample,accessingane-commercewebpage.To create this policy,youshouldanswersome questions suchas−
  - Shouldthepasswordbecomplexornot?
  - Whatageshouldtheusershave?
  - Maximum allowed tries orfails tolog in?
  - Whentheusershould bedeleted, activated,blocked?
- **Information Protection Policy** − This policy is to regulate access to information, hot toprocess information,how to storeand how itshouldbetransferred.
- **Remote Access Policy** − This policy is mainly for big companies where the user and theirbranches are outside their headquarters. It tells what should the users access, when theycanworkandon whichsoftwarelike SSH,VPN, RDP.
- **Firewall Management Policy** − This policy has explicitly to do with its management, whichports should be blocked, what updates should be taken, how to make changes in thefirewall,how longshouldbethelogsbe kept.
- **Special Access Policy** − This policy is intended to keep people under control and monitorthe special privileges in their systems and the purpose as to why they have it. Theseemployees can be team leaders, managers, senior managers, system administrators, andsuchhighdesignation basedpeople.
- **Network Policy** − This policy is to restrict the access of anyone towards the networkresource and make clear who all will access the network. It will also ensure whether thatperson should be authenticated or not. This policy also includes other aspects like, who willauthorize the new devices that will be connected with network? The documentation ofnetworkchanges.Webfiltersandthelevelsofaccess.Whoshouldhavewirelessconnectionandthetypeofauthentication,validityofconnectionsession?
- **Email Usage Policy** − This is one of the most important policies that should be donebecausemanyusersusetheworkemailforpersonalpurposesaswell.Asaresultinformation can leak outside. Some of the key points of this policy are the employeesshould know the importance of this system that they have the privilege to use. They shouldnot open any attachments that look suspicious. Private and confidential data should not besentviaanyencryptedemail.
- **Software Security Policy** − This policy has to do with the software's installed in the usercomputer and what they should have. Some of the key points of this policy are Software ofthe company should not be given to third parties. Only the white list of software's should beallowed,noothersoftware'sshouldbeinstalledinthecomputer.Warezandpiratedsoftware'shouldnotbeallowed.

1.4 AMODELFORNETWORKSECURITY

A model for much of what we will be discussing is captured, in very general terms, inFigure1.3.A messageistobe transferred fromonepartytoanotheracrosssomesortofInternetservice.

A security-related transformation on the information to be sent, Examples include theencryptionofthe message,whichscrambles themessage so thatitisunreadableby theopponent, and the addition of a code based on the contents of the message, which can be usedtoverifythe identityofthesender

Some secret information shared by the two principals and, it is hoped, unknown to theopponent. An example is an encryption key used in conjunction with the transformationtoscramblethe messagebeforetransmission andunscramble iton reception.

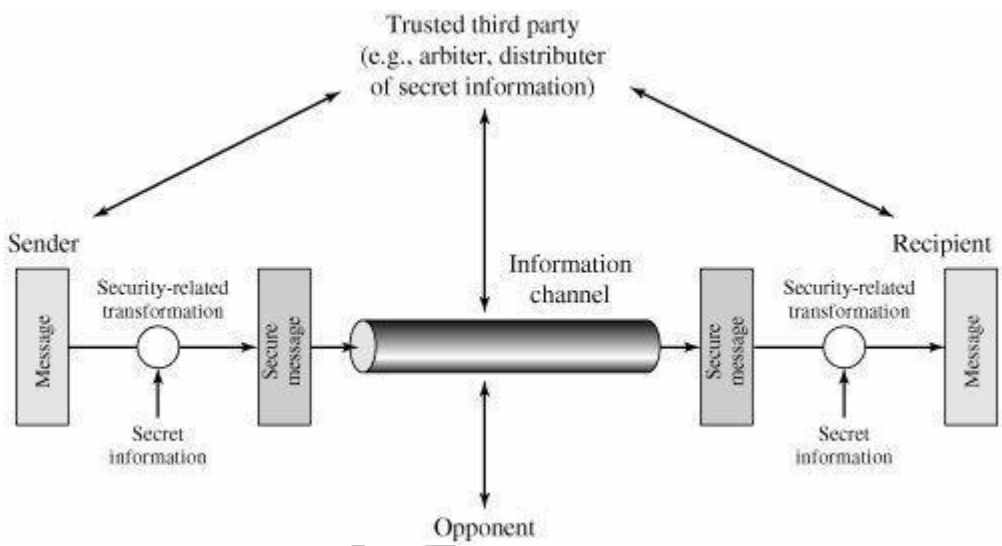


Figure1.3ModelforNetwork Security

Allthe techniquesforprovidingsecurityhavetwo components:
Thisgeneralmodelshowsthat therearefourbasictasksindesigninga particularsecurityservice:

1. Design an algorithm for performing the security-related transformation.Thealgorithmshouldbesuchthatanopponentcannotdefeatits purpose.

2. Generatethesecretinformationto beusedwith thealgorithm.

3. Develop methodsfor thedistribution andsharingofthe secretinformation.

4. Specify a protocol to be used by the two principals that makes use of the security algorithmandthe secretinformationto achieveaparticularsecurityservice

A general model of these other situations is illustrated by Figure 1.4, which reflects aconcern for protecting an information system from unwanted access. Most readers are familiarwith the concerns caused by the existence of hackers, who attempt to penetrate systems thatcan be accessed overa network. The hacker can be someone who,with no malign intent,simply gets satisfaction from breaking and entering a computer system. The intruder can be adisgruntled employee who wishes to do damage or a criminal who seeks to exploit computerassetsforfinancialgain(e.g.,obtainingcreditcardnumbersorperformingillegalmoneytransfers).

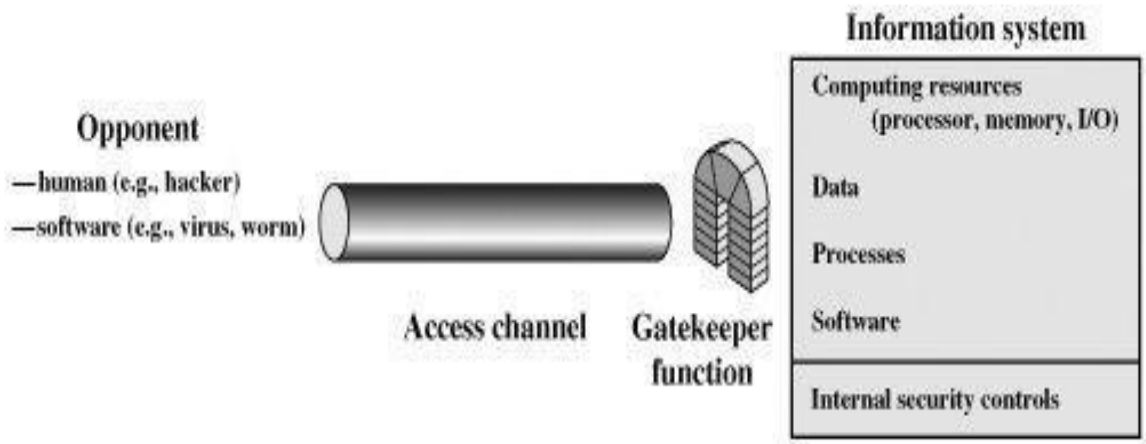


Figure1.4NetworkAccessSecurityModel

Another type of unwanted access is the placement in a computer system of logic that exploitsvulnerabilities in the system and that can affect application programs as well as utility programs,suchas editors and compilers.Programscan presenttwokindsofthreats:

• **Information access threats:** Intercept or modify data on behalf of users who should not haveaccesstothatdata.

• **Servicethreats:**Exploitserviceflaws in computersto inhibitusebylegitimateusers.

Viruses and worms are two examples of software attacks. Such attacks can be introduced into asystem by means of a disk that contains the unwanted logic concealed in otherwise usefulsoftware.

The security mechanisms needed to cope with unwanted access fall into twobroadcategories (see Figure 1.4).The first category might be termed a gatekeeper function. It includespassword-based login procedures that are designed to deny access to all but authorized usersand screening logic that is designed to detect and reject worms, viruses, and other similarattacks.Onceeitheranunwanteduserorunwantedsoftwaregainsaccess,

The second line of defense consists of a variety of internal controls that monitor activityand analyzestoredinformationinanattempttodetectthepresenceofunwantedintruders.

1.5 THEOSISECURITYARCHITECTURE

ITU-T Recommendation X.800, *Security Architecture for OSI*, defines such a systematicapproach. The OSIsecurity architecture is useful to managers as a way oforganizingthe taskof providing security. This architecture was developed as an international standard, computerand communications vendors have developed security features for their products and servicesthatrelatetothisstructureddefinitionofservicesandmechanisms.

TheOSIsecurity architecturefocusesonsecurityattacks,mechanisms, andservices. Thesecanbe definedbrieflyas

• **Security attack:**Anyactionthatcompromisesthesecurityofinformationownedbyanorganization.

• **Security mechanism:** A process (or a device incorporating such a process) thatis designedtodetect,prevent,orrecoverfroma security attack.

• **Security service:** A processing or communication service that enhances the security of thedata processing systems and the information transfers of an organization. The services areintended to counter security attacks, and they make use of one or more security mechanisms toprovide the service. In the literature, the terms *threat* and *attack* are commonly used to meanmore orlessthesamething.

•

Table1.1provides definitions takenfrom RFC2828,*InternetSecurityGlossary.*

Threat
A potential for violation of security, which exists when there is a circumstance, capability, action,or event that could breach security and cause harm. That is, a threat is a possible danger thatmightexploitavulnerability.
Attack
Anassaultonsystemsecuritythatderivesfromanintelligentthreat;thatis,anintelligentactthat is a deliberate attempt (especially in the sense of a method or technique) to evade securityservicesandviolatethesecuritypolicy ofasystem.

1.5.1 ATTACKS

The security attacks can be classified into two types' *passive attacks* and *active attacks.*A passive attack attempts to learn or make use ofinformation from the system but does notaffect system resources. An active attack attempts to alter system resources or affect theiroperation.

PassiveAttacks

Twotypesofpassive attacks aretherelease of messagecontentsandtrafficanalysis.

The**releaseofmessagecontents**iseasilyunderstood(Figure1.5a).Atelephoneconversation,anelectronicmailmessage,andatransferredfilemaycontainsensitiveorconfidential information. We would like to prevent an opponent from learning the contents ofthese transmissions.

A second type of passive attack, **traffic analysis**, is subtler (Figure 1.5b). Suppose thatwehadawayofmaskingthecontentsofmessagesorotherinformationtrafficsothatopponents,even ifthey captured the message,couldnotextractthe information from themessage. The common technique for masking contents is encryption. If we had encryptionprotectioninplace, anopponent mightstillbeabletoobserve thepatternofthesemessages.

Passive attacks are very difficultto detect,because they do notinvolve any alteration of thedata. Typically, the message traffic is not sent and received in an apparently normal fashion andthe sender nor receiver is aware that a third party has read the messages or observed the trafficpattern.

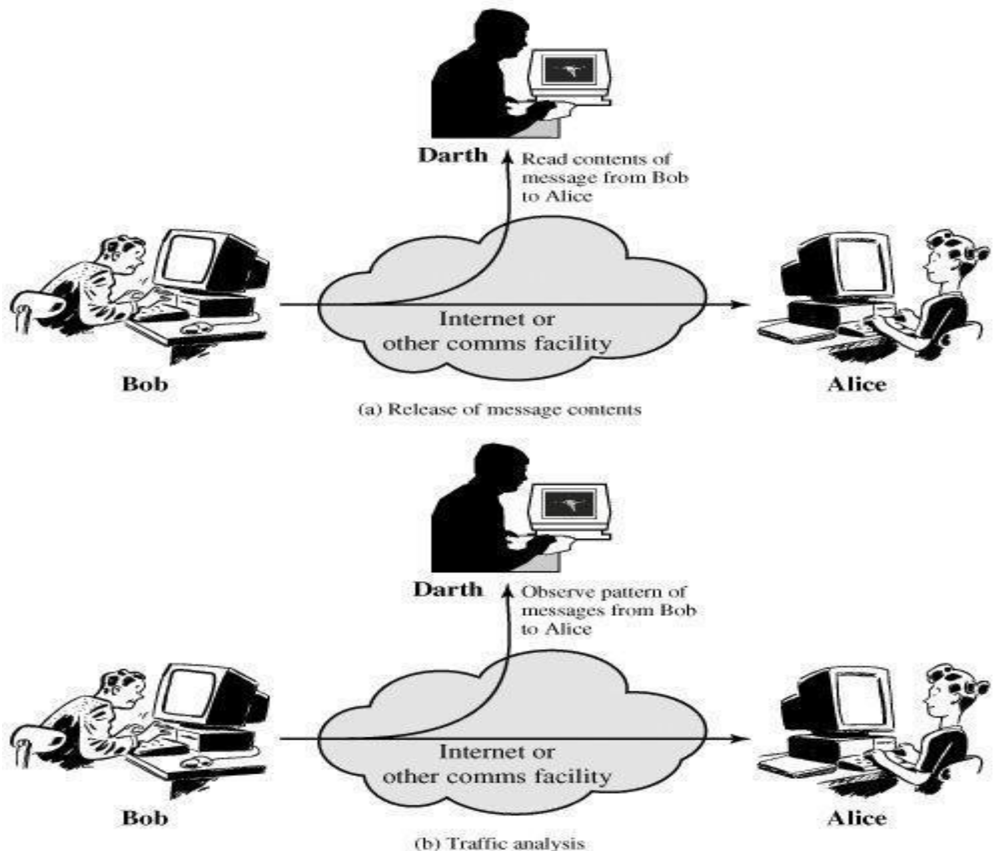


Figure1.5PassiveAttacks

ActiveAttacks

Active attacks involve some modification of the data stream or the creation of a false stream andcan be subdivided into four categories: masquerade, replay, modification of messages, anddenialofservice.

A **masquerade** takes place when one entity pretends to be a different entity (Figure 1.6a). Amasqueradeattackusuallyincludesoneoftheotherformsofactiveattack.Forexample,authentication sequences can be captured and replayed after a valid authentication sequencehas taken place, thus enabling an authorized entity with few privileges to obtain extra privilegesbyimpersonatingan entitythathas those privileges.

Replay involvesthepassivecaptureofadataunitanditssubsequentretransmissiontoproducean unauthorizedeffect(Figure1.6b).

Modification of messages simply means that some portion of a legitimate message is altered,or that messages are delayed or reordered, to produce an unauthorized effect (Figure 1.6c).
Forexample,amessagemeaning“AllowJohnSmithtoreadconfidentialfile*accounts*”ismodifiedto mean“Allow FredBrowntoreadconfidentialfile*account.*

The **denial of service** prevents or inhibits the normal use or management of communicationsfacilities(Figure1.6d).Thisattackmay haveaspecifictarget.

Activeattackspresent the oppositecharacteristicsofpassiveattacks.Whereaspassive attacksaredifficulttodetect,measuresareavailabletopreventtheirsuccess.

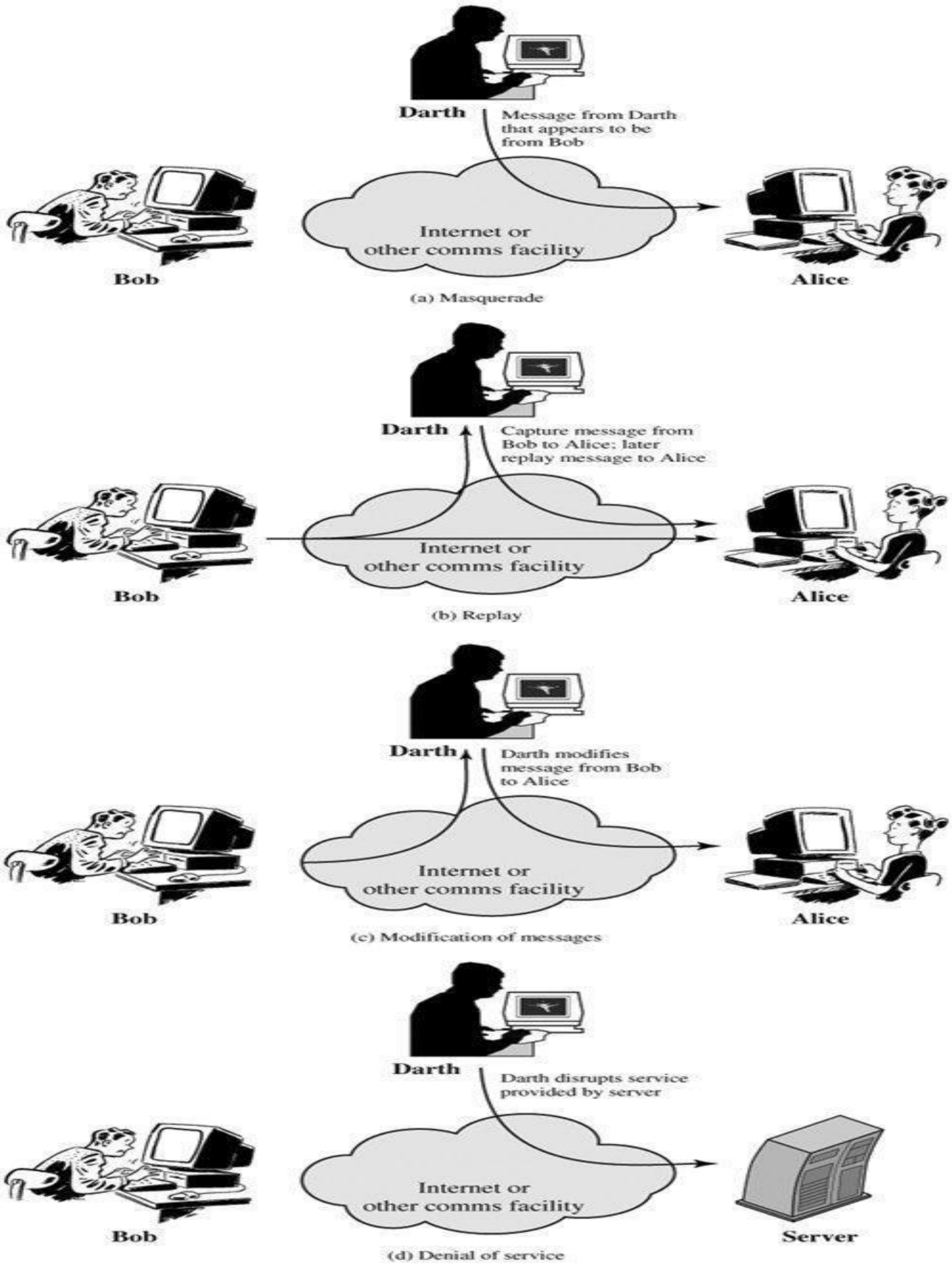


Figure1.6Active Attacks

1.5.2 SERVICES

X.800 defines a security service as a service that is provided by a protocol layer ofcommunicating open systems and that ensures adequate security of the systems or of datatransfers.Perhapsa clearerdefinition is found inRFC 2828, whichprovidesthe followingdefinition: a processing or communication service that is provided by a system to give a specifickind of protection to system resources; security services implement security policies and areimplementedby securitymechanisms.
X.800dividesthese services intofive categoriesandfourteenspecificservices(Table1.2)

Table1.2SecurityServices(X.800)

AUTHENTICATION

The assurance that the communicating entity is the one that it claims to be.

Peer Entity Authentication
Used in association with a logical connection to provide confidence in the identity of the entities connected.

Data-Origin Authentication
In a connectionless transfer, provides assurance that the source of received data is as claimed.

ACCESS CONTROL

The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).

DATA CONFIDENTIALITY

The protection of data from unauthorized disclosure.

Connection Confidentiality
The protection of all user data on a connection.

Connectionless Confidentiality
The protection of all user data in a single data block

Selective-Field Confidentiality
The confidentiality of selected fields within the user data on a connection or in a single data block.

Traffic-Flow Confidentiality
The protection of the information that might be derived from observation of traffic flows.

DATA INTEGRITY

The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).

Connection Integrity with Recovery
Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.

Connection Integrity without Recovery
As above, but provides only detection without recovery.

Selective-Field Connection Integrity
Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.

Connectionless Integrity
Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.

Selective-Field Connectionless Integrity
Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.

NONREPUDIATION

Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.

Nonrepudiation, Origin
Proof that the message was sent by the specified party.

Nonrepudiation, Destination
Proof that the message was received by the specified party.

1.5.3 MECHANISMS

Table1.3liststhesecuritymechanismsdefinedinX.800.Themechanismsaredividedintothose that are implemented in a specific protocol layer, such as TCP or an application-layerprotocol, and thosethatarenotspecific toany particularprotocollayerorsecurity service

.Table 1.3SecurityMechanisms(X.800)

SPECIFIC SECURITY MECHANISMS	PERVASIVE SECURITY MECHANISMS
May be incorporated into the appropriate protocol layer in order to provide some of the OSI security services.	Mechanisms that are not specific to any particular OSI security service or protocol layer.
Encipherment The use of mathematical algorithms to transform data into a form that is not readily intelligible. The transformation and subsequent recovery of the data depend on an algorithm and zero or more encryption keys.	**Trusted Functionality** That which is perceived to be correct with respect to some criteria (e.g., as established by a security policy).
Digital Signature Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery (e.g., by the recipient).	**Security Label** The marking bound to a resource (which may be a data unit) that names or designates the security attributes of that resource.
Access Control A variety of mechanisms that enforce access rights to resources.	**Event Detection** Detection of security-relevant events.
Data Integrity A variety of mechanisms used to assure the integrity of a data unit or stream of data units.	**Security Audit Trail** Data collected and potentially used to facilitate a security audit, which is an independent review and examination of system records and activities.
Authentication Exchange A mechanism intended to ensure the identity of an entity by means of information exchange.	**Security Recovery** Deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions.
Traffic Padding The insertion of bits into gaps in a data stream to frustrate traffic analysis attempts.	
Routing Control Enables selection of particular physically secure routes for certain data and allows routing changes, especially when a breach of security is suspected.	
Notarization The use of a trusted third party to assure certain properties of a data exchange.	

1.6 CLASSICALENCRYPTIONTECHNIQUES

Symmetricencryptionisaformofcryptosysteminwhichencryptionanddecryptionareperformedusingthesamekey.Itisalsoknownasconventionalencryption.

- Symmetric encryption transforms plaintext into ciphertext using a secret key and anencryption algorithm. Using the same key and a decryption algorithm, the plaintext isrecovered fromtheciphertext.
- Thetwotypesofattackonanencryptionalgorithmarecryptanalysis,basedonproperties of the encryption algorithm, and brute-force, which involves trying all possiblekeys.
- Traditional(precomputer)symmetriccipherssusesubstitutionand/ortranspositiontechniques. Substitutiontechniquesmapplaintextelements(characters,bits)intociphertext elements. Transposition techniques systematically transpose the positions ofplaintextelements.

- Rotormachinesaresophisticatedprecomputerhardwaredevicesthatusesubstitutiontechniques.
- Steganographyisatechniqueforhidingasecretmessagewithinalargeroneinsuchaway thatothers cannotdiscern thepresenceorcontents ofthehiddenmessage.

An original message is known as the **plaintext**, while the coded message is called the**ciphertext**. The process of converting from plaintext to ciphertext is known as **enciphering** or**encryption**; restoring the plaintext from the ciphertext is **deciphering** or **decryption**. The manyschemesusedfor encryptionconstitute the areaofstudyknownas**cryptography**.

Such a scheme is known as a **cryptographic system** or a **cipher**. Techniques used fordeciphering a message without any knowledge of the enciphering details fall into the area of**cryptanalysis**. Cryptanalysis iswhat the layperson calls “breaking the code” The areas ofcryptographyandcryptanalysistogetherare called**cryptology**.

1.6.1 SYMMETRICCIPHERMODEL

Asymmetric encryptionscheme hasfiveingredients (Figure1.7):

• **Plaintext:**Thisistheoriginalintelligiblemessageordata thatis fed intothealgorithmasinput.

• **Encryptionalgorithm:**Theencryptionalgorithmperformsvarioussubstitutionsandtransformations on theplaintext.

• **Secret key:** The secret key is also input to the encryption algorithm. The key is a valueindependent of the plaintext and of the algorithm. The algorithm will produce a different outputdependingonthespecifickeybeingusedatthetime.Theexactsubstitutionsandtransformations performedby the algorithmdependonthe key

• **Ciphertext:** This is the scrambled message produced as output. It depends on the plaintextandthesecretkey.Foragivenmessage,twodifferentkeyswillproducetwodifferentciphertexts.Theciphertextisanapparentlyrandomstreamofdataand,asitstands,isunintelligible.

• **Decryption algorithm:** This is essentially the encryption algorithm run in reverse. It takes theciphertextandthesecretkey and producestheoriginalplaintext.

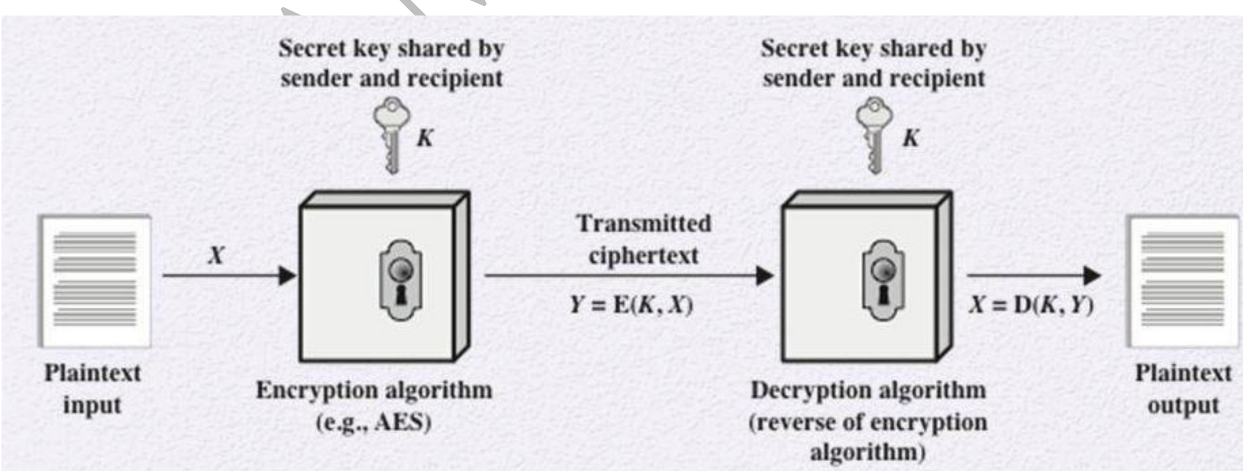


Figure1.7SimplifiedModelofSymmetricEncryption

Therearetwo requirements forsecure useofconventional encryption:

1. We need a strong encryption algorithm. At a minimum, we would like the algorithm to be suchthat an opponent who knows the algorithm and has access to one or more ciphertexts would beunable to decipher the ciphertext or figure out the key. This requirement is usually stated in astronger form: The opponent should be unable to decrypt ciphertext or discover the key even ifhe or she is in possession of a number of ciphertexts together with the plaintext that producedeachciphertext.

2. Sender and receiver must have obtained copies of the secret key in a secure fashion andmust keep the key secure.Ifsomeone can discoverthe key and knows the algorithm,allcommunicationusingthiskey isreadable.

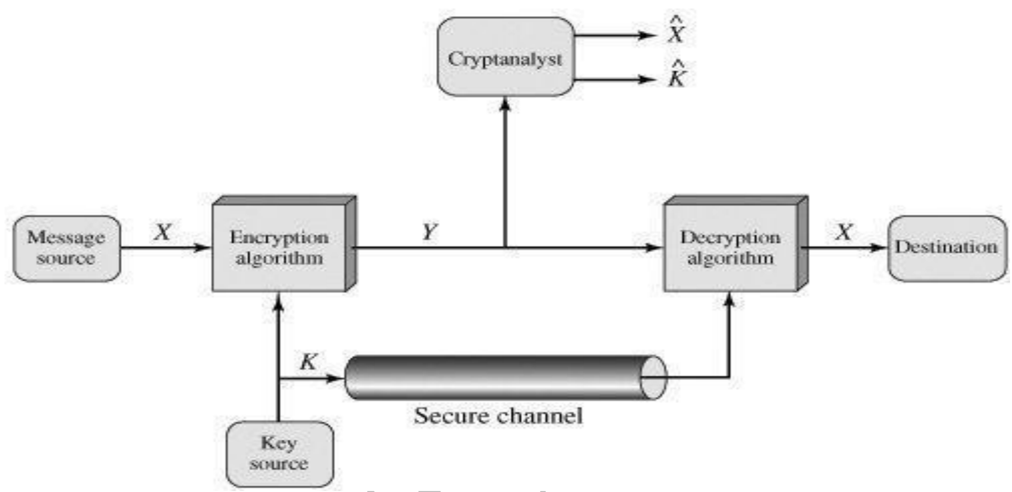


Figure1.8Model ofSymmetricCryptosystem

With the message X and the encryption key K as input, the encryption algorithmformsthe ciphertext Y=[Y1,Y2,.......Y_N] .We can write this as Y=E(K,X)This notation indicates that isproduced by using encryption algorithm E as a function of the plaintext X , with the specificfunctiondeterminedbythevalueofthe key K.

Theintendedreceiver,inpossessionof thekey,isable toinvertthetransformation:

X=D(K,Y)

Anopponent,observingYbutnothavingaccessKtoXor,mayattempttorecoverXorKorboth X and K. It is assumed that the opponent knows the encryption (E) and decryption (D)algorithms. If the opponent is interested in only this particular message, then the focus of theeffort is to recover X by generating a plaintext estimate X. Often, however, the opponent isinterested in being able to read future messages as well, in which case an attempt is made torecoverKbygeneratinganestimateK.

1.6.2 Cryptography

Cryptographicsystemsarecharacterizedalongthreeindependentdimensions:

Thetypeofoperationsused fortransformingplaintexttociphertext:
Allencryptionalgorithmsarebasedontwogeneralprinciples:substitution,inwhicheachelement in the plaintext (bit, letter, group of bits or letters) is mapped into another element, andtransposition, in which elements in the plaintext are rearranged. The fundamental requirement isthat no information be lost (that is, that all operations are reversible). Most systems, referred toas*productsystems*,involvemultiplestagesofsubstitutionsandtranspositions.

1. The number of keys used. If both sender and receiver use the same key, the system isreferred to as symmetric, single-key, secret-key, or conventional encryption. If the sender andreceiver use different keys, the system is referred to as asymmetric, two-key, or public-keyencryption.
2. The way in which the plaintext is processed. A *block cipher* processes the input one blockofelements ata time,producinganoutputblockforeachinputblock.A*stream cipher*processestheinputelementscontinuously,producingoutputoneelementatatime, asitgoesalong.
3. CryptanalysisandBrute-ForceAttack
Typically, the objective of attacking an encryption system is to recover the key in use rather thansimply to recover the plaintexts of a single ciphertext. There are two general approaches toattackingaconventional encryptionscheme:

• **Cryptanalysis:** Cryptanalytic attacks rely on the nature of the algorithm plusperhaps someknowledgeofthegeneralcharacteristicsoftheplaintextorevensomesampleplaintext-ciphertext pairs. This type of attack exploits the characteristics of the algorithm to attempt todeduceaspecificplaintextortodeducethekey beingused.
• **Brute-force attack**: The attacker tries every possible key on a piece of cipher text until anintelligible translation into plaintext is obtained. On average, half of all possible keys must betriedtoachievesuccess.

Table1.4summarizesthevarioustypesofcryptanalyticattacksbasedontheamountofinformation known to the cryptanalyst. The most difficult problem is presented when all that isavailableistheciphertextonly.

Table1.4TypesofAttacksonEncryptedMessages

Type of Attack	Known to Cryptanalyst
Ciphertext Only	• Encryption algorithm • Ciphertext
Known Plaintext	• Encryption algorithm • Ciphertext • One or more plaintext–ciphertext pairs formed with the secret key
Chosen Plaintext	• Encryption algorithm • Ciphertext • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key
Chosen Ciphertext	• Encryption algorithm • Ciphertext • Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key
Chosen Text	• Encryption algorithm • Ciphertext • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key • Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key

A**brute-forceattack**involvestryingeverypossiblekeyuntilanintelligibletranslationoftheciphertextintoplaintextisobtained.

1.6.3 SUBSTITUTIONTECHNIQUES

Thetwobasicbuildingblocksofallencryptiontechniquesaresubstitutionandtransposition. A substitution technique is one in which the letters of plaintext are replaced byother letters or by numbers or symbols.1 If the plaintext is viewed as a sequence of bits, thensubstitutioninvolvesreplacingplaintextbitpatternswithciphertextbitpatterns.

1.CaesarCipher

The earliest known, and the simplest, use of a substitution cipher was by Julius Caesar.The Caesar cipher involves replacing each letter of the alphabet with the letter standing threeplaces furtherdownthealphabet.Forexample,

plain:meet me after the toga party
cipher:PHHW PH DIWHU WKH WRJD SDUWB

Notethatthealphabetiswrappedaround,so thattheletterfollowingZisA.Wecandefine thetransformationby listingallpossibilities,as follows:
plain:abcdefghi j kl mnopqrstuvw xyz
cipher:D EF GH IJKL MNOPQRSTUVWXYZABC

Letusassign anumerical equivalenttoeachletter:
When letters are involved, the following conventions are used in this book. Plaintext is always inlowercase;ciphertextis inuppercase;keyvaluesareinitalicizedlowercase.

Letusassign anumerical equivalenttoeachletter:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12

n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

Then the algorithm can be expressed as follows. For each plaintext letter, substitute the ciphertextletter:

$C = \mathrm{E}(3,p) = (p+3) \bmod 26$

Ashiftmaybe ofany amount,sothatthegeneralCaesaralgorithmis

$C = \mathrm{E}(k,p) = (p + k) \bmod 26$

where takes on a valuein therange1 to 25.Thedecryption algorithmissimply

$p = \mathrm{D}(k,C) = (C - k) \bmod 26$

Ifitisknownthat agivenciphertextisa Caesar cipher, thena brute-forcecryptanalysisis easilyperformed: simply try all the 25 possible keys. Three important characteristics of this problemenabledustouseabruteforcecryptanalysis:

1. Theencryptionanddecryptionalgorithms areknown.

2. Thereareonly 25keys totry.

3. Thelanguageoftheplaintextisknownand easilyrecognizable.

	PHHW PH DIWHU WKH WRJD SDUWB
KEY	
1	oggv og chvgt vjg vqic rctva
2	nffu nf bgufs uif uphb qbsuz
3	meet me after the toga party
4	ldds ld zesdq sgd snfz ozqsx
5	kccr kc ydrcp rfc rmey nyprw
6	jbbq jb xcqbo qeb qldx mxoqv
7	iaap ia wbpan pda pkcw lwnpu
8	hzzo hz vaozm ocz ojbv kvmot
9	gyyn gy uznyl nby niau julns
10	fxxm fx tymxk max mhzt itkmr
11	ewwl ew sxlwj lzw lgys hsjlq
12	dvvk dv rwkvi kyv kfxr grikp
13	cuuj cu qvjuh jxu jewq fqhjo
14	btti bt puitg iwt idvp epgin
15	assh as othsf hvs hcuo dofhm
16	zrrg zr nsgre gur gbtn cnegl
17	yqqf yq mrfqd ftq fasm bmdfk
18	xppe xp lqepc esp ezrl alcej
19	wood wo kpdob dro dyqk zkbdi
20	vnnc vn jocna cqn cxpj yjach
21	ummb um inbmz bpm bwoi xizbg
22	tlla tl hmaly aol avnh whyaf
23	skkz sk glzkx znk zumg vgxze
24	rjjy rj fkyjw ymj ytlf ufwyd
25	qiix qi ejxiv xli xske tevxc

Figure1.9Brute-ForceCryptanalysisofCaesarCipher

2. MonoalphabeticCiphers

With only 25 possible keys, the Caesar cipher is far from secure.A dramatic increaseinthe key space can be achieved by allowingan arbitrary substitution.A **permutation** ofa finiteset of elements is an ordered sequence of all the elements of, with each element appearingexactlyonce.Forexample,ifS ={a,b,c},there aresixpermutationsof:

abc,acb,bac,bca,cab,cba

In general, there are n! permutations of a set of elements, because the first element canbechoseninoneof*n*ways,the secondin n-1ways,thethird inn-2 ways,andsoon.

Recall theassignmentfor theCaesarcipher:

plain:abcd efghIj klmn opqrstu vw xyz

cipher:D EF GH IJKL MNOPQRSTUVWXYZABC

If, instead, the “cipher” line can be any permutation of the 26 alphabetic characters, thenthere are 26! or greater than $4*10^{26}$ possible keys. This is 10 orders of magnitude greater thanthe key space for DES and would seem to eliminate brute-force techniques for cryptanalysis.Such an approach is referred to as a **monoalphabetic substitution cipher**, because a singlecipheralphabet(mappingfrom plainalphabettocipheralphabet)isusedpermessage.

Theciphertexttobesolvedis

UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZVUEPHZHMDZSHZO
WSFPAPPDTSVPQUZWYMXUZUHSXEPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMO
HMQ

As a first step, the relative frequency of the letters can be determined and compared to astandard frequency distribution for English, such as is shown in Figure 1.9. If the message werelong enough, this technique alone might be sufficient, but because this is a relatively shortmessage,wecannotexpectanexactmatch.Inanycase,therelativefrequenciesofthelettersintheciphertext(inpercentages) areasfollows:

P	13.33	H	5.83	F	3.33	B	1.67	C	0.00
Z	11.67	D	5.00	W	3.33	G	1.67	K	0.00
S	8.33	E	5.00	Q	2.50	Y	1.67	L	0.00
U	8.33	V	4.17	T	2.50	I	0.83	N	0.00
O	7.50	X	4.17	A	1.67	J	0.83	R	0.00
M	6.67								

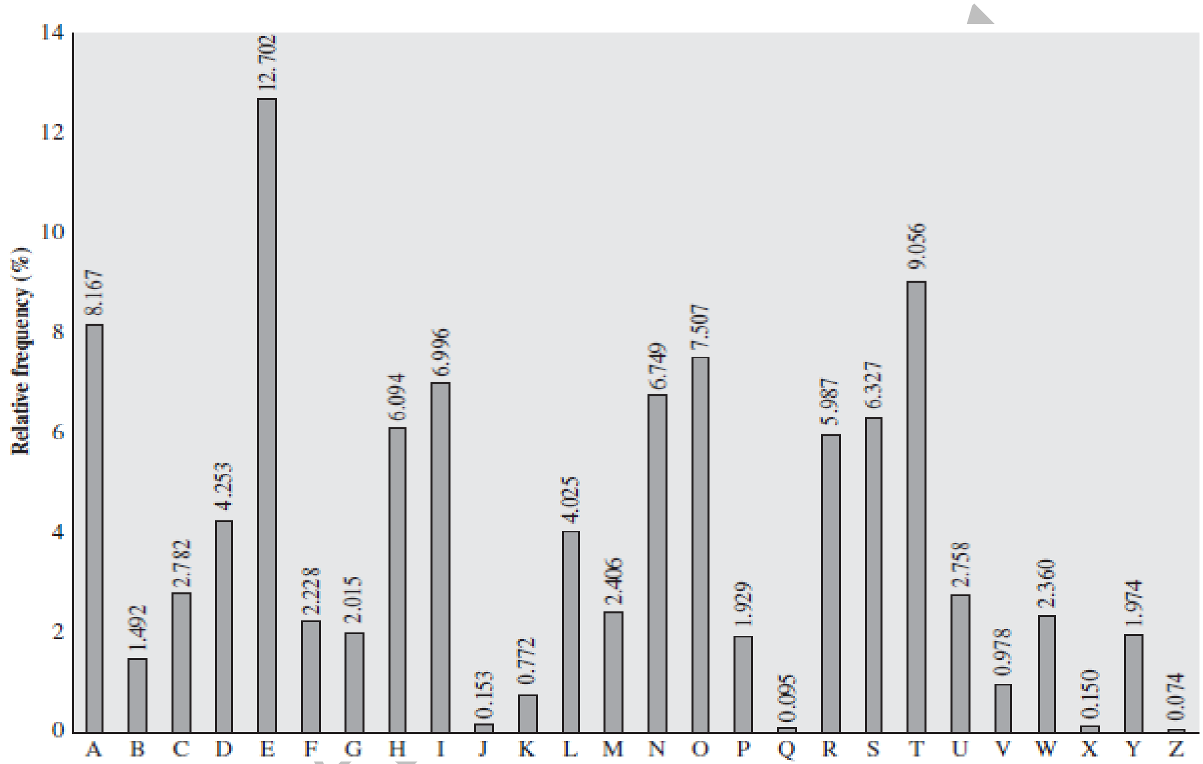


Figure1.10RelativeFrequenciesof LettersinEnglishText

That cipher letters P and Z are the equivalents of plain letters e and t, but it is not certainwhich is which. The letters S, U, O, M, and H are all of relatively high frequency and probablycorrespond to plain letters from the set {a, h, i, n, o, r, s}. The letters with the lowest frequencies(namely A,B,G,Y, I,J) arelikely includedintheset{b,j,k, q,v,x,z}.

A powerful tool is to look at the frequency of two-letter combinations, known as **digrams.**The most common such digram is th. In our ciphertext, the most common digram is ZW, whichappears three times. So we make the correspondence of Z with t and W with h. Then, by ourearlier hypothesis, we can equate P with e. Now notice that the sequence ZWP appears in theciphertext, and we can translate that sequence as “the.” This is the most frequent trigram (three-letter combination). Next, notice the sequence ZWSZ in the first line. We do not know that thesefour letters formacompleteword,butiftheydo,itisof theformth_t.Ifso,Sequates witha.

So far,then,wehave

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
 t a         e  e te  a that e e a        a
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
  e t   ta t ha e ee  a e  th    t  a
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
 e  e e tat  e   the   t
```

Onlyfourlettershavebeenidentified,butalreadywehavequiteabitofthemessage.Continued analysis of frequencies plus trial and error should easily yield a solution from thispoint.Thecompleteplaintext,withspacesaddedbetweenwords,follows:

it was disclosed yesterday that several informal
butdirect contacts have been made with
politicalrepresentativesofthevietconginmoscow

Monoalphabeticciphersareeasytobreakbecausetheyreflectthe frequencydataoftheoriginal alphabet. A countermeasure is to provide multiple substitutes, known as homophones,for asingleletter.

3. PlayfairCipher

The best-known multiple-letter encryption cipher is the Playfair, which treats digrams inthe plaintext as single units and translates these units into ciphertext digrams. The Playfairalgorithmisbasedontheuseofa5×5matrixoflettersconstructedusingakeyword.Hereisanexample,solvedby LordPeterWimsey inDorothy Sayers‟s*HaveHisCarcase*

M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

In this case, the keyword is *monarchy*. The matrix is constructed by filling in the letters of thekeyword (minus duplicates) from left to right and from top to bottom, and then filling in theremainder of the matrix with the remaining letters inalphabetic order. The letters I and J countas oneletter. Plaintextisencryptedtwolettersatatime,accordingtothefollowingrules:

1. Repeating plaintext letters that are in the same pair are separated with a fillerletter, such asx,sothatballoonwouldbetreated asba lxloon.

2. Two plaintext letters that fall in the same row of the matrix are each replaced by the letter totheright,withthe firstelementofthe row circularly followingthelast.Forexample,arisencryptedasRM.

3. Two plaintext letters that fall in the same column are each replaced by the letter beneath, withthe topelementofthecolumncircularlyfollowingthelast.Forexample,muisencryptedasCM.

4. Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row andthe column occupied by the other plaintext letter. Thus, hs becomes BP and ea becomes IM (orJM,asthe enciphererwishes).

The Playfair cipher is a great advance over simple monoalphabetic ciphers. For one thing,whereas there are only 26 letters, there are 26 × 26 = 676 digrams, so that identification ofindividual digrams is more difficult. Furthermore, the relative frequencies of individual lettersexhibit a much greater range than that ofdigrams, making frequency analysis much moredifficult. For these reasons, the Playfair cipher was for a long time considered unbreakable. Itwas used as the standard field system by the British Army in World War I and still enjoyedconsiderableusebythe U.S.ArmyandotherAllied forcesduringWorldWarII.

4. HillCipher

Another interesting multiletter cipher is the Hill cipher, developed by the mathematicianLester Hill in 1929. Define the inverse $\mathbf{M}^{-1}$of a square matrix **M** by the equation **$\mathbf{M}(\mathbf{M}^{-1})= \mathbf{M}^{-1}\mathbf{M}=\mathbf{I}$,**where**I** istheidentitymatrix.**I**isasquarematrixthatisallzerosexcept for ones alongthemaindiagonalfromupperleft tolowerright.Theinverse ofamatrixdoes not always exist,butwhen

$$\mathbf{A} = \begin{pmatrix} 5 & 8 \\ 17 & 3 \end{pmatrix} \qquad \mathbf{A}^{-1} \bmod 26 = \begin{pmatrix} 9 & 2 \\ 1 & 15 \end{pmatrix}$$

$$\mathbf{AA}^{-1} = \begin{pmatrix} (5 \times 9) + (8 \times 1) & (5 \times 2) + (8 \times 15) \\ (17 \times 9) + (3 \times 1) & (17 \times 2) + (3 \times 15) \end{pmatrix}$$
$$= \begin{pmatrix} 53 & 130 \\ 156 & 79 \end{pmatrix} \bmod 26 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

itdoes,it satisfiestheprecedingequation.For example,

To explain how the inverse of a matrix is computed, we begin by with the concept ofdeterminant. For any square matrix (m × m), the **determinant** equals the sum of all the productsthat can be formed by taking exactly one element from each row and exactly one element fromeachcolumn,withcertainoftheproducttermsprecededbya minussign.Fora2 ×2matrix,

$$\begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix}$$

Thedeterminant is$k_{11}k_{22}$-$k_{12}k_{21}$.For a3×3matrix,thevalue ofthedeterminantis .$k_{11}k_{22}k_{33}+k_{21}k_{32}k_{13}+k_{31}k_{12}k_{23}-k_{31}k_{22}k_{13}-k_{21}k_{12}k_{33}-k_{11}k_{32}k_{23}$.Ifasquarematrix**A** hasanonzerodeterminant,thentheinverseofthematrixiscomputedas$[A^{-1}]_{ij}=(\det A)^{-1}(-1)^{i+j}(D_{ij})$where$(D_{ij})$isthesubdeterminantformedbydeletingthejthrowandtheithcolumnof**A**,det(**A**)isthe determinant of**A**,and $(\det A)^{-1}$isthemultiplicative inverseof(det**A**)mod26.
Continuingour example,

$$\det \begin{pmatrix} 5 & 8 \\ 17 & 3 \end{pmatrix} = (5 \times 3) - (8 \times 17) = -121 \bmod 26 = 9$$

Wecanshowthat9^{-1}mod26=3,because9×3=27mod26=1.Therefore,we computetheinverseof**A**as

$$\mathbf{A} = \begin{pmatrix} 5 & 8 \\ 17 & 3 \end{pmatrix}$$

$$\mathbf{A}^{-1} \bmod 26 = 3\begin{pmatrix} 3 & -8 \\ -17 & 5 \end{pmatrix} = 3\begin{pmatrix} 3 & 18 \\ 9 & 5 \end{pmatrix} = \begin{pmatrix} 9 & 54 \\ 27 & 15 \end{pmatrix} = \begin{pmatrix} 9 & 2 \\ 1 & 15 \end{pmatrix}$$

THEHILLALGORITHMThisencryptionalgorithmtakesmsuccessiveplaintextlettersandsubstitutesforthemmciphertextletters.Thesubstitutionisdeterminedbymlinearequationsin

$$c_1 = (k_{11}p_1 + k_{12}p_2 + k_{13}p_3) \bmod 26$$
$$c_2 = (k_{21}p_1 + k_{22}p_2 + k_{23}p_3) \bmod 26$$
$$c_3 = (k_{31}p_1 + k_{32}p_2 + k_{33}p_3) \bmod 26$$

whicheachcharacterisassignedanumericalvalue(a=0,b=1,...z=25).Form=3,thesystemcanbedescribed as

This canbeexpressedintermsofrowvectors andmatrices:

$$(c_1\ c_2\ c_3) = (p\ \ p_2\ \ p_3)\begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \bmod 26$$

or

C =**PK**mod26

where**C**and**P**are rowvectors oflength3 representingtheplaintextandciphertext,and **K**isa3×3matrixrepresentingtheencryption key.Operationsare performedmod26.Forexample,considertheplaintext"paymoremoney"and usetheencryptionKey

$$\mathbf{K} = \begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix}$$

The first three letters of the plaintext are represented by the vector (15 0 24). Then $(15\ 0\ 24)\mathbf{K} = (303\ 303\ 531) \bmod 26 = (17\ 17\ 11) = \text{RRL}$. Continuing in this fashion, the ciphertext for the entire plaintext is RRLMWBKASPDH.

Decryption requires using the inverse of the matrix **K**. We can compute $\det \mathbf{K} = 23$, and therefore, $(\det \mathbf{K})^{-1} \bmod 26 = 17$. We can then compute the inverse as

$$\mathbf{K}^{-1} = \begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix}$$

This is demonstrated as

$$\begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix}\begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix} = \begin{pmatrix} 443 & 442 & 442 \\ 858 & 495 & 780 \\ 494 & 52 & 365 \end{pmatrix} \bmod 26 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

It is easily seen that if the matrix $\mathbf{K}^{-1}$ is applied to the ciphertext, then the plaintext is recovered.

In general terms, the Hill system can be expressed as

$$\mathbf{C} = \mathrm{E}(\mathbf{K}, \mathbf{P}) = \mathbf{PK} \bmod 26$$
$$\mathbf{P} = \mathrm{D}(\mathbf{K}, \mathbf{C}) = \mathbf{CK}^{-1} \bmod 26 = \mathbf{PKK}^{-1} = \mathbf{P}$$

As with Playfair, the strength of the Hill cipher is that it completely hides single-letterfrequencies.Indeed,with Hill,the use ofa largermatrix hides more frequencyinformation.Thus,a3×3Hill cipherhidesnotonlysingle-letter but also two-letterfrequencyinformation.

Consider this example. Suppose that the plaintext “hillcipher” is encrypted using a HillciphertoyieldtheciphertextHCRZSSXNSP.Thus,weknowthat(78)Kmod26=(72)1111)Kmod26=(17 25);andsoon.Usingthefirsttwoplaintext-ciphertextpairs,wehave

$$\begin{pmatrix} 7 & 2 \\ 17 & 25 \end{pmatrix} = \begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix} \mathbf{K} \bmod 26$$

The inverseof**X**can be computed

$$\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}^{-1} = \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix}$$

so

$$\mathbf{K} = \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix} \begin{pmatrix} 7 & 2 \\ 17 & 25 \end{pmatrix} = \begin{pmatrix} 549 & 600 \\ 398 & 577 \end{pmatrix} \bmod 26 = \begin{pmatrix} 3 & 2 \\ 8 & 5 \end{pmatrix}$$

Thisresultisverifiedbytestingtheremainingplaintext-ciphertextpairs.

5. OneTimePadCipher(or)VernamCipher

It is an unbreakable cryptosystem, described by Frank Miller in 1882, the one-time padwas reinvented by Gilbert Vernam in 1917 and it was later improved by the US Amry MajorJoseph. It represents the message as a sequence of 0s and 1s. This can be accomplished bywriting all numbers in binary, for example, or by using ASCII. The key is a random sequence of0‟sand 1‟sofsamelengthas themessage.

Once a key is used, it is discarded and never used again. The system can be expressedas follows:

$$C_i = P_i \oplus K_i$$

C_i - i^{th} binary digit of cipher text

P_i - i^{th} binary digit of plaintext

K_i - i^{th} binary digit of key

$\oplus$ – exclusive OR opearaiton

Thus the cipher text is generated by performing the bitwise XOR of the plaintext and thekey.Decryptionusesthesame key.Because oftheproperties ofXOR,decryptionsimplyinvolvesthesamebitwiseoperation:

$$P_i = C_i \oplus K_i$$

Example

Alice wishes to send the message “HELLO” to Bob. If key material begins with “XMCKL” and themessage is “HELLO”, then use Vernam One Time Pad to Decrypt and Show the EncryptionProcess.

MESSAGE	H	E	L	L	O
POSITION	7	4	11	11	14
KEY	X	M	C	K	L
POSITION	23	12	2	10	11

OTPEncryption

H	E	L	L	O	Message
7 (H)	4 (E)	11 (L)	11 (L)	14 (O)	Message
23 (X)	12 (M)	2 (C)	10 (K)	11 (L)	Key
30	16	13	21	25	Message+Key
4 (E)	16 (Q)	13 (N)	21 (V)	25 (Z)	Message+Key (mod 26)
E	Q	N	V	Z	Ciphertext

Note: Ifanumberislarger than 25, thentheremainderafersubtractionof 26istakenin ModularArithmetic fashion

OTPDecryption

E	Q	N	V	Z	Ciphertext
4 (E)	16 (Q)	13 (N)	21 (V)	25 (Z)	Ciphertext
23 (X)	12 (M)	2 (C)	10 (K)	11 (L)	Key
-19	4	11	11	14	Ciphertext-Key
7 (H)	4 (E)	11 (L)	11 (L)	14 (O)	Ciphertext-Key(mod26)
H	E	L	L	O	Message

Note: Ifanumberisnegativethen 26isaddedtomakethenumberpositive

Example

Encryption

Plaintext is 00101001 and the key is 10101100, we obtain the ciphertext is,

Plaintext	00101001
Key	10101100
Ciphertext	10000101

Decryption

Ciphertext	10000101
Key	10101100
Plaintext	00101001

Advantages

- Encryption method is completelyunbreakable fora cipher-textonly knownattack
- Chosen Plaintext (or) Ciphertext attacks is not

possibleDisadvantages

- It requiresaverylongkeywhich isexpensive to produceandexpensive totransmit.
- Onceakey isuseditisdangerous toreuseitforsecondmessage.

6. PolyalphabeticCiphers

Another way to improve on the simple monoalphabetic technique is to use differentmonoalphabetic substitutionsas one proceeds through the plaintextmessage. Thegeneralname for this approach is **polyalphabetic substitution cipher**. All these techniques have thefollowingfeaturesincommon:

1. Aset ofrelatedmonoalphabetic substitution rulesisused.
2. Akeydetermineswhich particular rule is chosenforagiventransformation.

VIGEN`ERE CIPHER The best known, and one of the simplest, polyalphabetic ciphers is theVigenère cipher. In this scheme, the set of related monoalphabetic substitution rules consists ofthe26Caesarcipherswithshiftsof0through25.Eachcipherisdenotedbyakeyletter,whichis the ciphertext letter that substitutes for the plaintext letter a. Thus, a Caesar cipher with a shiftof3 isdenotedby thekeyvalue.

Express the Vigenère cipher in the following manner. Assume a sequence of plaintextletters and a key consisting of the sequence of letters, where typically < .The sequence ofciphertextlettersiscalculatedas follows

$$
\begin{aligned}
C = C_0, C_1, C_2, \ldots, C_{n-1} &= \mathrm{E}(K, P) = \mathrm{E}[(k_0, k_1, k_2, \ldots, k_{m-1}), (p_0, p_1, p_2, \ldots, p_{n-1})] \\
&= (p_0 + k_0) \bmod 26, (p_1 + k_1) \bmod 26, \ldots, (p_{m-1} + k_{m-1}) \bmod 26, \\
&\quad (p_m + k_0) \bmod 26, (p_{m+1} + k_1) \bmod 26, \ldots, (p_{2m-1} + k_{m-1}) \bmod 26, \ldots
\end{aligned}
$$

Thus, the first letter of the key is added to the first letter of the plaintext, mod 26, the secondletters are added, and so on through the first letters of the plaintext. For the next letters of theplaintext, the key letters are repeated. This process continues until all of the plaintext sequenceisencrypted.A generalequationoftheencryptionprocessis

$C_i = (p_i + k_{i \bmod m}) \bmod 26$

Decryptionisageneralization ofEquation

$p_i = (C_i - k_{i \bmod m}) \bmod 26$

To encrypt a message, a key is needed that is as long as the message. Usually, the keyisarepeatingkeyword.Forexample,ifthekeywordis*deceptive*,themessage“wearediscoveredsav eyourself”isencryptedas

key:deceptivedeceptivedeceptiveplaint
ext:wearediscoveredsaveyourself
ciphertext:ZICVTWQNGRZGVTWAVZHCQYGLMGJ

1.6.4 TRANSPOSITIONTECHNIQUES

All the techniques examined so far involve the substitution of a ciphertext symbol for aplaintext symbol. A very differentkind of mapping is achieved by performingsome sort ofpermutation on the plaintext letters. This technique is referred to as a transposition cipher. Thesimplest such cipher is the **rail fence** technique, in which the plaintext is written down as asequence of diagonals and then read off as a sequence of rows. For example, to encipher themessage“meetmeafterthetogaparty”witharailfenceofdepth2,wewritethefollowing:

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

Theencryptedmessageis

MEMATRHTGPRYETEFETEOAAT

This sort of thing would be trivial to cryptanalyze. A more complex scheme is to write themessage in a rectangle, row by row, and read the message off, column by column, but permutethe order of the columns. The order of the columns then becomes the key to the algorithm. Forexample,

```
Key:           4 3 1 2 5 6 7
Plaintext:     a t t a c k p
               o s t p o n e
               d u n t i l t
               w o a m x y z
Ciphertext:    TTNAAPTMTSUOAODWCOIXKNLYPETZ
```

Thus, in this example, the key is 4312567.To encrypt, start with the column thatislabeled 1, in this case column 3.Write down all the letters in that column. Proceed to column 4,which is labeled 2, then column 2, then column 1, then columns 5, 6, and 7.A pure transpositioncipher is easily recognized because it has the same letter frequencies as the original plaintext.For the type of columnar transposition just shown, cryptanalysis is fairly straightforward andinvolves laying out the ciphertext in a matrix and playing around with column positions. Digramandtrigram frequencytablescanbeuseful.

The transposition cipher can be made significantly more secure by performing more thanonestageoftransposition.Theresultisamorecomplexpermutationthatisnoteasilyreconstructed. Thus,ifthe foregoingmessageisreencryptedusingthesamealgorithm,

```
Key:          4 3 1 2 5 6 7
Input:        t t n a a p t
              m t s u o a o
              d w c o i x k
              n l y p e t z
Output:       NSCYAUOPTTWLTMDNAOIEPAXTTOKZ
```

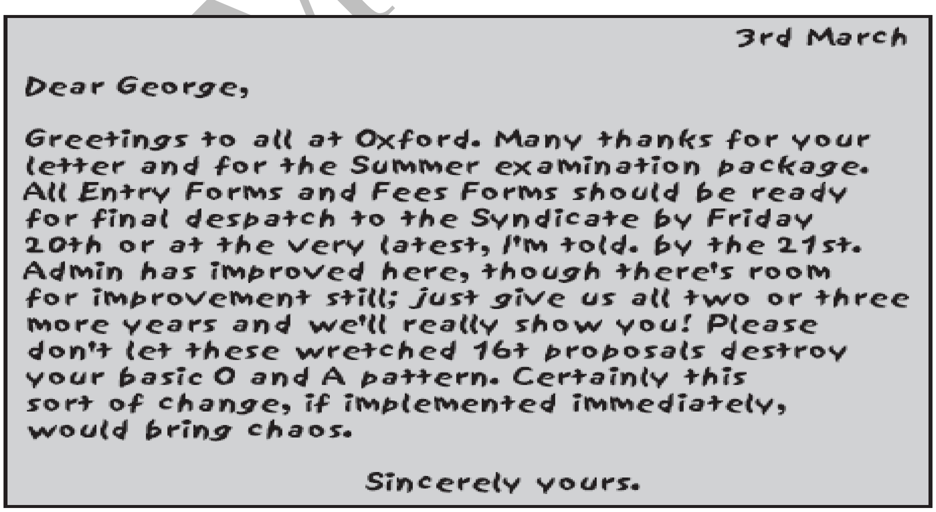
To visualize the result of this double transposition, designate the letters in the originalplaintextmessagebythenumbersdesignatingtheirposition.Thus,with28lettersinthemessage ,theoriginalsequence ofletters is

0102030405060708091011121314
1516171819202122 232425262728
Afterthefirst transposition,we have
0310172404111825 020916230108
1522051219260613 202707142128
MEMATRHTGPRYETEFETEOAAT

1.7 STEGANOGRAPHY

Aplaintextmessagemaybehiddeninoneoftwoways.Themethodsof**steganography** conceal the existence of the message, whereas the methods of cryptographyrenderthemessageunintelligibletooutsiders byvarioustransformationsofthetext.

A simple form of steganography, but one that is time-consuming to construct, is one inwhich an arrangement of words or letters within an apparently innocuous text spells out the realmessage. For example, the sequence of first letters of each word of the overall message spellsout the hidden message. Figure shows an example in which a subset of the words of the overallmessageisused to convey thehidden message.


3rd March

Dear George,

Greetings to all at Oxford. Many thanks for your letter and for the Summer examination package. All Entry Forms and Fees Forms should be ready for final despatch to the Syndicate by Friday 20th or at the very latest, I'm told. by the 21st. Admin has improved here, though there's room for improvement still; just give us all two or three more years and we'll really show you! Please don't let these wretched 16+ proposals destroy your basic O and A pattern. Certainly this sort of change, if implemented immediately, would bring chaos.

Sincerely yours.

Various other techniques have been used historically; some examples are the following:**Character marking:** Selected letters of printed or typewritten text are overwritten in pencil. Themarks are ordinarily notvisibleunlessthepaperisheldatanangletobrightlight.

Invisibleink: Anumber ofsubstancescanbeusedforwriting but leave novisible traceuntilheatorsomechemicalisappliedto thepaper.

Pinpunctures: Smallpinpuncturesonselectedlettersareordinarilynotvisibleunlessthepaperisheld upinfrontofa light.

Typewritercorrectionribbon: Usedbetweenlinestypedwithablackribbon,theresultsoftypingwithth ecorrection tapearevisibleonlyunderastronglight

Steganography has a number of drawbacks when compared to encryption. It requires a lot ofoverhead to hide a relatively few bits of information, although using a scheme like that proposedin the preceding paragraph may make it more effective. Also, once the system is discovered, itbecomesvirtuallyworthless.Thisproblem,too,canbeovercomeiftheinsertionmethoddependson somesortofkey.

Theadvantageofsteganographyisthatitcanbeemployedbypartieswhohavesomething to lose should the fact of their secret communication (not necessarily the content) bediscovered. Encryption flags traffic as important or secret or may identify the sender or receiverassomeonewithsomethingtohide.

1.8 Foundationsofmoderncryptography

Modern encryption is the key to advanced computer and communication security. Thisstream of cryptography is completely based on the ideas of mathematics such as number theoryandcomputational complexity theoryaswellasconceptsofprobability.

CharacteristicsofModernCryptography

Therearefourmajorcharacteristicsthatseparatemoderncryptographyfromtheclassicalappro ach.

Table1.5DifferencesbetweenTraditionalEncryptionandModern Encryption

TraditionalEncryption	**ModernEncryption**
Formakingciphertext,manipulation isdone inthecharactersoftheplaintext	Formakingciphertext, operationsare performedonbinary bitsequence
Thewholeoftheecosystemis requiredtocommunicate confidentiality	Here,only thepartieswhowanttoexecute securecommunicationpossessthesecretke y
These areweakerascompared tomodernencryption	The encryption algorithm formed by thisencryption technique isstrongeras comparedtotraditionalencryptionalgorithms
It believes intheconceptofsecuritythrough obscurity	Itssecuritydependsonthepublicly known mathematicalalgorithm

ContextofCryptography

Cryptology,thestudyofcryptosystems,canbesubdividedintotwobranches−

- Cryptography
- Cryptanalysis

Cryptography

Cryptographyistheartandscienceofmakingacryptosystemthatiscapableofproviding information security. Cryptography deals with the actual securing of digital data. Itrefers to the design of mechanisms based on mathematical algorithms that provide fundamentalinformationsecurityservices.

Cryptanalysis

The art and science of breaking the cipher text is known as cryptanalysis. Cryptanalysisis the sister branch of cryptography and they both co-exist. The cryptographic process results inthecipher text for transmissionor storage. It involvesthestudy ofcryptographic mechanismwith the intention to break them.Cryptanalysisisalso used during the design ofthe newcryptographic techniquesto testtheirsecurity strengths.

Note − Cryptography concerns with the design of cryptosystems, while cryptanalysis studies thebreakingofcryptosystems.

TypesofModernCryptography

Differentalgorithmshavecomeupwithpowerfulencryptionmechanismsincorporatedinthem.It gaverise totwonewwaysofencryption mechanismfordata security.Theseare:

- Symmetrickeyencryption
- Asymmetrickeyencryption

Key

Itcanbeanumber,word,phrase,oranycodethatwillbeusedforencryptingaswellas decrypting anyciphertextinformationtoplaintext and viceversa.

Symmetric and asymmetric key cryptography is based on the number of keys and theway thesekeyswork.Letusknow aboutboth ofthemindetails:

Symmetrickeyencryption

Symmetrickeyencryptiontechniqueusesastraightforwardmethodofencryption.Hence, this is the simpler among these two practices. In the case of symmetric key encryption,the encryption is done through only one secret key, whichis knownas"Symmetric Key", andthis key remainstoboththeparties.

The same key is implemented for both encodings as well as decoding the information.So, the key is used first by the sender prior to sending the message, and on the receiver side,that key isused to deciphertheencodedmessage.

One of the good old examples of this encryption technique is Caesar's Cipher. Modernexamples and algorithms that use the concept of symmetric key encryption are RC4, QUAD,AES,DES,Blowfish,3DES,etc.

AsymmetricKeyEncryption

Asymmetric Encryption is another encryption method that uses two keys, which is a newand sophisticated encryption technique. This is because it integrates two cryptographic keys forimplementingdata security.These keysare termedasPublicKey andPrivateKey.

The"publickey",asthenameimplies,isaccessibletoallwhowanttosendanencrypted message. The other is the "private key" that is kept secure by the owner of that publickeyorthe one whoisencrypting.

Encryption of information is done through public key first, with the help of a particularalgorithm. Then the private key, which the receiver possesses, will use to decrypt that encryptedinformation.Thesame algorithmwillbeusedinbothencodings aswellasdecoding.

ExamplesofasymmetrickeyencryptionalgorithmsareDiffie-HellmanandRSAalgorithm.

SecurityServicesofCryptography

- Confidentialityofinformation.
- DataIntegrity.
- Authentication.
    - Message authentication.
    - Entityauthentication.
- Non-repudiation.

CryptographyPrimitives

Cryptographyprimitivesarenothingbutthetoolsandtechniquesin Cryptographythatcanbeselectivelyusedtoprovideasetofdesiredsecurityservices−

- Encryption
- Hashfunctions
- Message Authenticationcodes (MAC)
- DigitalSignatures

Thefollowingtableshowstheprimitivesthatcanachieveaparticularsecurityserviceontheirown
.

Table1.6PrimitivesandSecurityService

Primitives / Service	Encryption	Hash Function	MAC	Digital Signature
Confidentiality	Yes	No	No	No
Integrity	No	Sometimes	Yes	Yes
Authentication	No	No	Yes	Yes
Non Reputation	No	No	Sometimes	Yes

1.8.1 PerfectSecurity

PerfectSecrecy(orinformation-theoreticsecure)meansthattheciphertextconveysnoinformationaboutthecontentoftheplaintext ....However,partofbeingprovablysecureisthat youneedas muchkeymaterialasyouhaveplaintexttoencrypt.

1.8.2 InformationTheory

Information theory studies the quantification, storage, and communication of information.Itwasoriginallyproposedby ClaudeShannon in1948tofindfundamentallimitson signalprocessingand communicationoperationssuchasdata compression.

Its impact has been crucial to the success of the Voyager missions to deep space, theinvention of the compact disc, the feasibility of mobile phones, the development of the Internet,thestudyof linguistics andofhumanperception,theunderstandingof blackholes,andnumerousotherfields.Thefieldisattheintersectionofmathematics, statistics,computerscience,physics,neurobiology,informationengineering,andelectricalengineerin g.

Thetheoryhasalsofoundapplicationsinotherareas, including statisticalinference, naturallanguageprocessing, cryptography, neurobiology, humanvision,theevolution and function of molecular codes (bioinformatics), model selection in statistics, thermalphysics, quantum computing, linguistics, plagiarism detection,pattern recognition, and anomalydetection.

Important sub-fields of information theory include source coding, algorithmic complexitytheory, algorithmicinformationtheory, information-theoreticsecurity, Greysystemtheory andmeasures ofinformation.

Applicationsoffundamentaltopicsofinformationtheoryinclude losslessdatacompression (e.g. ZIPfiles), lossydatacompression (e.g. MP3s and JPEGs),and channelcoding(e.g.forDSL).

Information theory is used in information retrieval, intelligence gathering, gambling, andeveninmusicalcomposition.

Akeymeasureininformationtheoryis entropy.Entropyquantifiestheamountofuncertainty involved in the value of a random variable or the outcome of a random process. Forexample, identifying the outcome of a fair coin flip (with two equally likely outcomes) providesless information (lower entropy) than specifying the outcome from a roll of a die (with six equallylikely outcomes). Some other important measures in information theory are mutual information,channelcapacity,errorexponents,andrelativeentropy.

1.8.3 ProductCryptosystems

A product cipher combines two or more transformations in a manner intending that theresultingcipherismoresecurethantheindividualcomponentstomakeitresistanttocryptanalysis.

Theproductciphercombines a sequence of simple transformations suchas substitution (S-box), permutation (P-box),and modulararithmetic.Fortransformationinvolvingreasonablenumberofn messagesymbols,bothofthe foregoingciphersystems(the S-boxandP-box)areby themselveswanting.

The combination could yield a cipher system more powerful than either one alone. Thisapproachofalternativelyapplyingsubstitutionandpermutationtransformationhasbeenusedby IBM in the Lucifer cipher system, and has become the standard for national data encryptionstandards such as the Data Encryption Standard and the Advanced Encryption Standard.Aproduct cipher that uses only substitutions and permutations is called a SP-network. Feistelciphersareanimportant classofproductciphers.

1.9 CRYPTANALYSIS

Cryptanalysisis the artoftryingto decrypt the encrypted messages without the use ofthe key that was used to encrypt the messages. Cryptanalysis uses mathematical analysis &algorithms to decipherthe ciphers.

Thesuccess ofcryptanalysisattacks depends

- Amountoftimeavailable
- Computingpoweravailable
- Storagecapacityavailable

Thefollowingisa list ofthe commonlyused Cryptanalysisattacks;

Brute force attack- this type of attack uses algorithms that try to guess all the possiblelogical combinations of the plaintext which are then ciphered and compared against the originalcipher.

Dictionary attack– this type ofattack uses a wordlist in order to find a match of eithertheplaintextorkey.Itismostlyusedwhentryingtocrackencrypted passwords.

Rainbowtableattack–thistypeofattackcomparestheciphertextagainstpre-computedhashestofind matches.

OtherAttacksusingCryptanalysis

Known-PlaintextAnalysis(KPA): Attackerdecryptsciphertextwithknownpartialplaintext.

Chosen-Plaintext Analysis (CPA): Attacker uses ciphertext that matches arbitrarilyselectedplaintextviathesamealgorithmtechnique.

Ciphertext-OnlyAnalysis(COA): Attackerusesknownciphertext collections.

Man-in-the-Middle (MITM) Attack: Attack occurs when two parties use message or keysharingforcommunicationviaachannelthatappearssecurebutisactuallycompromised. Attacker employs this attack for the interception of messages that passthroughthecommunicationschannel.HashfunctionspreventMITMattacks.

Adaptive Chosen-Plaintext Attack (ACPA): Similar to a CPA, this attack uses chosenplaintextandciphertextbasedondatalearned frompastencryptions.

UNITIISYMMETRICKEYCRYPTOGRAPHY

MATHEMATICSOFSYMMETRICKEYCRYPTOGRAPHY:AlgebraicStructures-Modulararithmetic - Euclid's Algorithm - Congruence and Matrices - Groups, Rings, Fields - Finite fields - SYMMETRIC KEY CIPHERS: SDES - Block cipher principles of DES - Strength of DES - Differential and Linear Cryptanalysis - Block cipher design principles - Block cipher mode ofoperation-Evaluationcriteria ofAES -AdvancedEncryptionStandard - RC4-KeyDistribution

2.1 ALGEBRAICSTRUCTURES

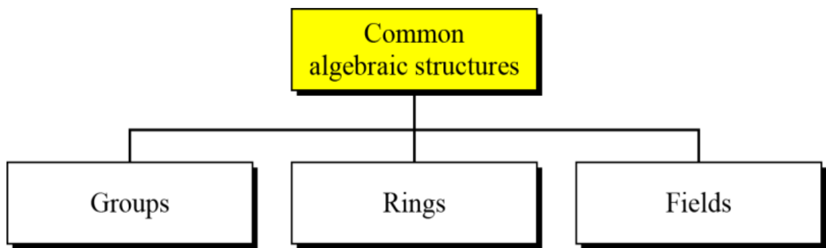


Figure2.1CommonAlgebraicStructures

2.1.1Groups,Rings, Fields

Groups,rings,andfieldsarethefundamentalelementsofabranchofmathematicsknown asabstractalgebra,ormodernalgebra.

Groups

A group G , sometimes denoted by {G,*} ,is a set of elements with a binaryoperation denoted by * that associates to each ordered pair (a,b) of elements G in anelement(a*b)in,such thatthe followingaxiomsareobeyed:

(A1) Closure: **IfaandbbelongtoG,thena*bisalsoinG.**
(A2)Associative:a*(b*c)=(a*b)*cforalla,b,,inG.
(A3) Identityelement: **Thereis anelementeinGsuch thata*e=e*a=aforallinG .**
(A4)Inverseelement: **ForeachainG,thereisanelement a'inG suchthata*a'=a'*a=e.**

If a group has a finite number of elements, it is referred to as a **finite group**, andthe **order** of the group is equal to the number of elements in the group. Otherwise, thegroupisan**infinite group**.

Agroupissaidtobe **abelian**ifit satisfiesthefollowingadditionalcondition:
(A5) Commutative: a*b= b*afor allab,inG.

CYCLIC GROUP: A group is cyclic if every element of G is a power a^k (k is aninteger) of a fixed element a£ G .The element is a said to generate the group G or to be ageneratorof
G.Acyclicgroupis always abelianandmaybefinite orinfinite.

Rings

AringR,sometimesdenotedby{R,+,X},isasetofelementswithtwobinary operations,calledadditionandmultiplication,suchthatforalla,b,c,inRthefollowingaxiomsareobeyed

(A1–A5) R is an abelian group with respect to addition; that is, R satisfies axioms A1 through A5. For the case of an additive group, we denote the identity element as 0 and the inverse of a as $-a$.

(M1) Closure under multiplication: If a and b belong to R, then ab is also in R.

(M2) Associativity of multiplication: $a(bc) = (ab)c$ for all a, b, c in R.

(M3) Distributive laws: $a(b + c) = ab + ac$ for all a, b, c in R.
$(a + b)c = ac + bc$ for all a, b, c in R.

Aringissaidtobe**commutative**ifitsatisfiesthefollowingadditionalcondition:

(M4) Commutativity of multiplication: $ab = ba$ for all a, b in R.

Next, we define an integral domain, which is a commutative ring that obeys the followingaxioms

(M5) Multiplicative identity: There is an element 1 in R such that $a1 = 1a = a$ for all a in R.

(M6) No zero divisors: If a, b in R and $ab = 0$, then either $a = 0$ or $b = 0$.

Fields

AfieldF,sometimesdenotedby{F,+,X},isasetofelementswithtwobinary operations,calledadditionandsubtraction,suchthatforalla,b,c,inFthefollowingaxiomsareobeyed

(A1–M6) F is an integral domain; that is, F satisfies axioms A1 through A5 and M1 through M6.

(M7) Multiplicative inverse: For each a in F, except 0, there is an element a^{-1} in F such that $aa^{-1} = (a^{-1})a = 1$.

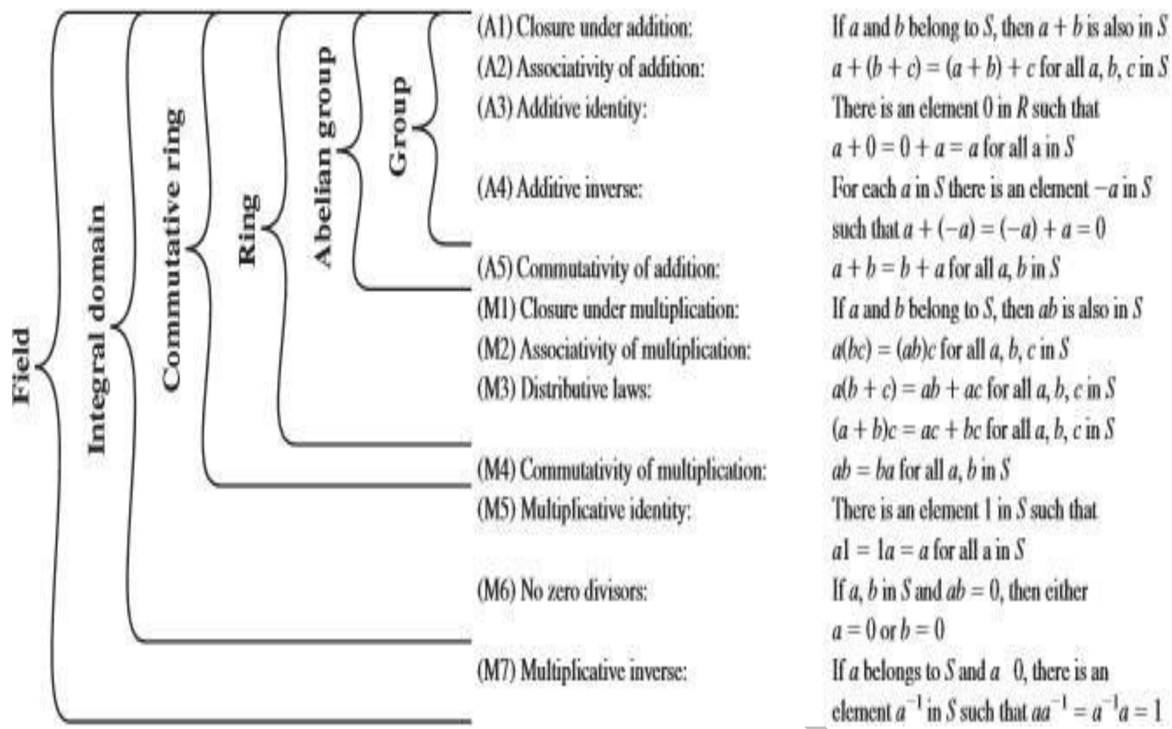


Figure2.2Groups,Ring and Field

2.2 MODULARARITHMETIC

If is an integer and n is a positive integer, we define a mod n to be the remainderwhen a is divided by n.The integer n is called the modulus. Thus, for any integera, wecanrewriteEquationasfollows

$$a = qn + r \qquad 0 \le r < n;\ q = \lfloor a/n \rfloor$$

$$a = \lfloor a/n \rfloor \times n + (a \bmod n)$$

$$11 \bmod 7 = 4; \qquad -11 \bmod 7 = 3$$

Two integers a and b are said to be **congruent modulo** $\boldsymbol{n}$, if $(a \bmod n) = (b \bmod n)$. This is written as $a = b \pmod{n}$.[2]

$$73 \equiv 4 \pmod{23}; \qquad 21 \equiv -9 \pmod{10}$$

Note that if $a = 0 \pmod{n}$, then $n | a$.

ModularArithmeticOperations

A kind of integer arithmetic that reduces all numbers to one of a fixed set [0,….,n-1] forsome number n. Any integer outside this range is reduced to one in this range by takingtheremainderafterdivisionby n.
Modular arithmeticexhibitsthefollowingproperties

1. $[(a \bmod n) + (b \bmod n)] \bmod n = (a + b) \bmod n$
2. $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$
3. $[(a \bmod n) \times (b \bmod n)] \bmod n = (a \times b) \bmod n$

We demonstrate the first property. Define $(a \bmod n) = r_a$ and $(b \bmod n) = r_b$. Then we can write $a = r_a + jn$ for some integer j and $b = r_b + kn$ for some integer k. Then

$$\begin{aligned}(a + b) \bmod n &= (r_a + jn + r_b + kn) \bmod n \\ &= (r_a + r_b + (k + j)n) \bmod n \\ &= (r_a + r_b) \bmod n \\ &= [(a \bmod n) + (b \bmod n)] \bmod n\end{aligned}$$

The remaining properties are proven as easily. Here are examples of the three properties:

Table2.1ArithmeticModulo8

$11 \bmod 8 = 3$; $15 \bmod 8 = 7$
$[(11 \bmod 8) + (15 \bmod 8)] \bmod 8 = 10 \bmod 8 = 2$
$(11 + 15) \bmod 8 = 26 \bmod 8 = 2$

$[(11 \bmod 8) - (15 \bmod 8)] \bmod 8 = -4 \bmod 8 = 4$
$(11 - 15) \bmod 8 = -4 \bmod 8 = 4$

$[(11 \bmod 8) \times (15 \bmod 8)] \bmod 8 = 21 \bmod 8 = 5$
$(11 \times 15) \bmod 8 = 165 \bmod 8 = 5$

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

(a) Addition modulo 8

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

(b) Multiplication modulo 8

w	$-w$	w^{-1}
0	0	—
1	7	1
2	6	—
3	5	3
4	4	—
5	3	5
6	2	—
7	1	7

(c) Additive and multiplicative inverses modulo 8

2.3 EUCLID'SALGORITHM

Oneofthebasictechniquesofnumbertheoryisthe Euclideanalgorithm,whichisa simple procedure for determining the greatest common divisor of two positive integers.First, we need a simple definition: Two integers are relatively prime if their only commonpositiveintegerfactoris1.

GreatestCommonDivisor

Recall that nonzero b is defined to be a divisor ofa if a =mb for some m, where a,b,and m are integers. We will use the notation gcd(a , b) to mean the greatest commondivisor of a and b .The greatest common divisor of a and b is the largest integer thatdividesbothaandb .Wealsodefinegcd(0,0)=0.

Algorithm

The Euclid's algorithm (or Euclidean Algorithm) is a method forefficiently finding the greatestcommondivisor (GCD) of two numbers. The GCD oftwo integers X and Y is the largest number that divides both of X and Y (without leaving aremainder).

For every non-negative integer, a and any positive integer bgcd(a,b)=gcd (b,amodb)

AlgorithmEuclids(a, b)

α=
aβ=b
while(β>0)
Rem=αmodβ
α=β
β =
Remreturnα

Stepsfor
AnotherMethoda=
q1b+r1;0<r1 <b
b=
q2r1+r2;0<r2<r1r1=q3r2
+r3;0<r3<r2
rn-2=qnrn-1+rn; 0 < rn < rn-
1rn-1 =q1rn +0
d =gcd (a,b)
=rnExample1:

gcd (55,22)=gcd(22,55mod 22)
=gcd(22,11)
=gcd (11,22mod11)
=gcd(11,0)

gcd(55,22)is 11

Example 2:

gcd(30,50)=gcd(50,30mod 50)
=gcd(50,30)
=gcd (30,50mod30)
=gcd(30,20)
=gcd(20,30mod20)
=gcd(20,10)
=gcd (10,20 mod10)
=gcd(10,0)

gcd (30, 50) is
10AnotherMetho
d

Tofind gcd (30,50)

50	=1x30+20	gcd(30,20)
30	=1x20+10	gcd(20,10)
20	=1x10+10	gcd(10,10)
10	=1 x10+0	gcd (10,0)

Therefore,gcd(30,50)=10

Example 3:

gcd(1970,1066)=gcd (1066,1970 mod 1066)
=gcd(1066,904)
=gcd(904,1066mod 904)
=gcd(904,162)
=gcd(162,904mod162)
=gcd(162,94)
=gcd(94,162mod 94)
=gcd(94,68)
=gcd(68,94mod68)
=gcd(68,26)
=gcd(26,68mod26)
=gcd(26,16)
=gcd(16,26mod16)
=gcd(16,10)
=gcd(10,16mod10)
=gcd(10,6)
=gcd(6,10mod6)

=gcd(6,4)

=gcd(4,6mod4)

=gcd(4,2)

=gcd(2,4mod2)

=gcd(2,0)

gcd(1970,1066)is 2

AnotherMethod

Tofind gcd(1970,1066)

1970	=1 x1066+904	gcd(1066,904)
1066	=1 x904+162	gcd (904,162)
904	=5 x162+94	gcd(162,94)
162	=1x94+68	gcd(94,68)
94	=1x68+26	gcd(68,26)
68	=2x26+16	gcd(26,16)
26	=1x16+10	gcd(16,10)
16	=1 x10+6	gcd(10,6)
10	=1 x6+4	gcd(6,4)
6	=1 x4+2	gcd(4,2)
4	=2 x2+0	gcd(2,0)

Therefore,gcd(1970,1066)=2

ExtendedEuclideanAlgorithm

ExtendedEuclideanAlgorithmisanefficientmethodoffindingmodularinverseofaninteger.

Euclid'salgorithmcanbeimprovedtogivenotjustgcd (a,b),butalsousedtofindthemultiplicativeinverseofanumberwiththe modularvalue.

Example 1

Find theMultiplicative inverse of17mod

4317-1mod43

17* X=

mod43X=17-

1mod43

43=17*2 +9

17=9*1+8

9=8*1+1

Rewrite the above

equation9+8(-1)=1→(1)

17+9(-1)=8→(2)

43+17(-2)=9→(3)

Substitution

subequ2 inequ 1

(1)→9+8(-1)=1[Sub 17+9(-1)=8]

9+(17+9(-1))(-1) =1

9+17(-1)+9(1)=1

17(-1)+9(2)=1→(4)

Nowsubequ(3)inequ

(4)43+17(-2)=9→(3)

17(-1)+(43+17(-2))(2)=1

17(-1)+43(2)+17(-4)=1

17(-5)+43(2)=1→(5)

Here -5 is the multiplicative inverse of 17. But inverse cannot be negative17-1mod43=-5 mod 43=38

So, 38 is the multiplicative inverse of 17.Checking,17*X≡1 mod 43

17*38 ≡1 mod 43

646≡1 mod43(15*43 =645)

Example 2

Find the Multiplicative inverseof1635mod 261635-1mod26

1635=26 (62)+23

26=23 (1)+3

23=3(7)+2

3=2(1)+1

Rewriting the above equation3+2(-1)=1→(1)

23+3(-7)=2→(2)

26+23(-1)=3→(3)

1635+26(-62)=23→(4)

Substitution

subequ (2) inequ(1)

(2)=>23+3(-7)=2

3+2(-1)=1

3+(23+3(-7))(-1) =1

3+23(-1)+3(7)=1

3(8)+23(-1)=1 →(5)

subequ (3) inequ(5)

26+23(-1)=3→(3)

(26+23(-1))(8)+23 (-1)=1

26(8)+23 (-8)+23(-1)=1

26(8)+23 (-9)=1→(6)

Subequ (4)inequ(6)

1635+26(-62)=23→(4)

26(8)+(1635 +26 (-62))(-9)=1

26(8)+1635 (-9)+26 (558)=1

1635(-9)+26 (566)=1→(7)

From equ (7) -9isinverse of1635,

butnegativecannotbeinverse.1635-1mod26 =-9 mod 26 =17

So,theinverseof1635is17.Che

cking, 1635*X≡1mod26

1635*17≡1 mod 26

27795≡1 mod26 (1069*26=27794)

2.4 CONGRUENCEANDMATRICES

PropertiesofCongruences

Congruenceshavethefollowingproperties:

1. $a \equiv b \pmod{n}$ if $n|(a - b)$.
2. $a \equiv b \pmod{n}$ implies $b \equiv a \pmod{n}$.
3. $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ imply $a \equiv c \pmod{n}$.

To demonstrate the first point, if $n|(a - b)$, then $(a - b) = kn$ for some k. So we can write $a = b + kn$. Therefore, $(a \bmod n)$ = (remainder when $b + kn$ is divided by n) = (remainder when b is divided by n) = $(b \bmod n)$.

$23 \equiv 8 \pmod{5}$	because	$23 - 8 = 15 = 5 \times 3$
$-11 \equiv 5 \pmod{8}$	because	$-11 - 5 = -16 = 8 \times (-2)$
$81 \equiv 0 \pmod{27}$	because	$81 - 0 = 81 = 27 \times 3$

The remaining points are as easily proved.

Matrices

Matrixisarectangulararrayinmathematics,arrangedinrowsandcolumns ofnumbers,symbolsorexpressions.

Amatrixwillberepresentedwiththeirdimensionsaslxmwhereldefinestherowandmdefin esthe columns *m* columns

l rows

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{lm} \end{bmatrix}$$

ExamplesofMatrices

1. RowMatrix
2. ColumnMatrix
3. SquareMatrix
4. ZeroMatrixes
5. IdentityMatrix

$$\begin{bmatrix} 2 & 1 & 5 & 11 \end{bmatrix}$$

Row matrix

$$\begin{bmatrix} 2 \\ 4 \\ 12 \end{bmatrix}$$

Column matrix

$$\begin{bmatrix} 23 & 14 & 56 \\ 12 & 21 & 18 \\ 10 & 8 & 31 \end{bmatrix}$$

Square matrix

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

0

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

I

2.5 FINITEFIELDS

FINITEFIELDSOF THE FORMGF(p)

The finite field of order is generally written ; GF stands for Galois field,in honor ofthemathematicianwho firststudiedfinite fields

Finite FieldsofOrderp

Foragivenprime,,we definethefinitefieldoforder,,asthesetofintegerstogetherwiththe arithmeticoperationsmodulo.

The simplest finite field is GF(2). Its arithmetic operations are easily summarized:

Addition

+	0	1
0	0	1
1	1	0

Multiplication

×	0	1
0	0	0
1	0	1

Inverses

w	$-w$	w^{-1}
0	0	—
1	1	1

In this case, addition is equivalent to the exclusive-OR (XOR) operation, and multiplication is equivalent to the logical AND operation.

Finding the Multiplicative Inverse in It is easy to find the multiplicative inverse of anelement in for small values of .You simply construct a multiplication table, such as showninTable2.2b,andthedesiredresultcanbereaddirectly.However,forlargevaluesof,thisap proachisnotpractical.p p GF(p)GF(p)

Table2.2Arithmetic inGF(7)

+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

(a) Addition modulo 7

×	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

(b) Multiplication modulo 7

w	$-w$	w^{-1}
0	0	—
1	6	1
2	5	4
3	4	5
4	3	2
5	2	3
6	1	6

(c) Additive and multiplicative inverses modulo 7

2.5.1 PolynomialArithmetic

We are concerned with polynomials in a single variable and we can distinguishthree classes of polynomial arithmetic. • Ordinary polynomial arithmetic, using the basicrules of algebra. • Polynomial arithmetic in which the arithmetic on the coefficients isperformedmodulo
;thatis,thecoefficientsare in .

Polynomial arithmetic in which the coefficients are in ,and the polynomials aredefinedmodulo apolynomialwhosehighestpowerissome integer**.**

OrdinaryPolynomialArithmetic

A polynomial ofdegree(integer)isanexpressionoftheform

A **polynomial** of degree n (integer $n \geq 0$) is an expression of the form

$$f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^{n} a_i x^i$$

where the a_i are elements of some designated set of numbers S, called the **coefficient set**, and $a_n \neq 0$. We say that such polynomials are defined over the coefficient set S.

A zero-degree polynomial is called a **constant polynomial** and is simply an element of the set of coefficients. An nth-degree polynomial is said to be a **monic polynomial** if $a_n = 1$.

In the context of abstract algebra, we are usually not interested in evaluating a polynomial for a particular value of x [e.g., $f(7)$]. To emphasize this point, the variable x is sometimes referred to as the **indeterminate**.

Additionandsubtractionareperformedbyaddingorsubtractingcorrespondingcoefficients.Thus ,if

$$f(x) = \sum_{i=0}^{n} a_i x^i; \qquad g(x) = \sum_{i=0}^{m} b_i x^i; \qquad n \geq m$$

then addition is defined as

$$f(x) + g(x) = \sum_{i=0}^{m} (a_i + b_i)x^i + \sum_{i=m+1}^{n} a_i x^i$$

and multiplication is defined as

$$f(x) \times g(x) = \sum_{i=0}^{n+m} c_i x^i$$

where

$$c_k = a_0 b_k + a_1 b_{k-1} + \cdots + a_{k-1} b_1 + a_k b_0$$

As an example, let $f(x) = x^3 + x^2 + 2$ and $g(x) = x^2 - x + 1$, where S is the set of integers. Then

$$\begin{aligned} f(x) + g(x) &= x^3 + 2x^2 - x + 3 \\ f(x) - g(x) &= x^3 + x + 1 \\ f(x) \times g(x) &= x^5 + 3x^2 - 2x + 2 \end{aligned}$$

PolynomialArithmeticwithCoefficientsin

Let us now consider polynomials in which the coefficients are elements of somefield F; we refer to this as a polynomial over the field F. In that case, it is easy to show thatthe set of such polynomials is a ring, referred to as a polynomial ring. That is, if weconsider each distinct polynomial to be an element ofthe set, then that set is a ring8when polynomial arithmetic is performed on polynomials over a field, then division ispossible. Note that this does not mean that exact division is possible. Let us clarify thisdistinction. Within a field, given two elements and, the quotient is also an element of thefield.However,givenaringthatisnota field,inRa/b baZp

$$\begin{array}{r} x^3 + x^2 \qquad + 2 \\ + \ (x^2 - x + 1) \\ \hline x^3 + 2x^2 - x + 3 \end{array}$$

(a) Addition

$$\begin{array}{r} x^3 + x^2 \qquad + 2 \\ - \ (x^2 - x + 1) \\ \hline x^3 \qquad + x + 1 \end{array}$$

(b) Subtraction

$$\begin{array}{r} x^3 + x^2 \qquad + 2 \\ \times \ (x^2 - x + 1) \\ \hline x^3 + x^2 \qquad + 2 \\ -x^4 - x^3 \qquad - 2x \\ x^5 + x^4 \qquad + 2x^2 \\ \hline x^5 \qquad + 3x^2 - 2x + 2 \end{array}$$

(c) Multiplication

$$\begin{array}{r} x + 2 \\ x^2 - x + 1 \,\big)\, x^3 + x^2 \qquad + 2 \\ x^3 - x^2 + x \\ \hline 2x^2 - x + 2 \\ 2x^2 - 2x + 2 \\ \hline x \end{array}$$

(d) Division

Figure2.3ExamplesofPolynomialArithmetic

A polynomial over a field is called irreducible if and only if cannot be expressed asa product of two polynomials, both over, and both of degree lower than that of. By analogyto integers,anirreduciblepolynomialisalsocalledaprimepolynomial.

2.6 SYMMETRICKEYCIPHERS

Symmetricciphers usethesame cryptographickeys forboth encryption ofplaintext and decryption of ciphertext. They are faster than asymmetric ciphers and allowencryptinglargesetsofdata.However,theyrequiresophisticatedmechanismstosecurely distribute the secretkeysto both parties

Definition A symmetric cipher defined over (K, M, C), where:

- K - a set of all possible keys,
- M - a set of all possible messages,
- C - a set of all possible ciphertexts

is a pair of efficient algorithms (E, D), where:

- E: K × M -> C
- D: K × C -> M

such that for every m belonging to M, k belonging to K there is an equality:

- D(k, E(k, m)) = m (the consistency rule)

➡ *Function E is often randomized*

➡ *Function D is always deterministic*

Typesofkeysare usedin symmetric keycryptography

Symmetricencryption(figure2.4) usesasinglekeythatneedstobesharedamongthepeoplewhoneedtoreceivethemessagewhile asymmetricalencryptionusesapairofpublickeyandaprivatekeytoencryptanddecryptmessage swhencommunicating.

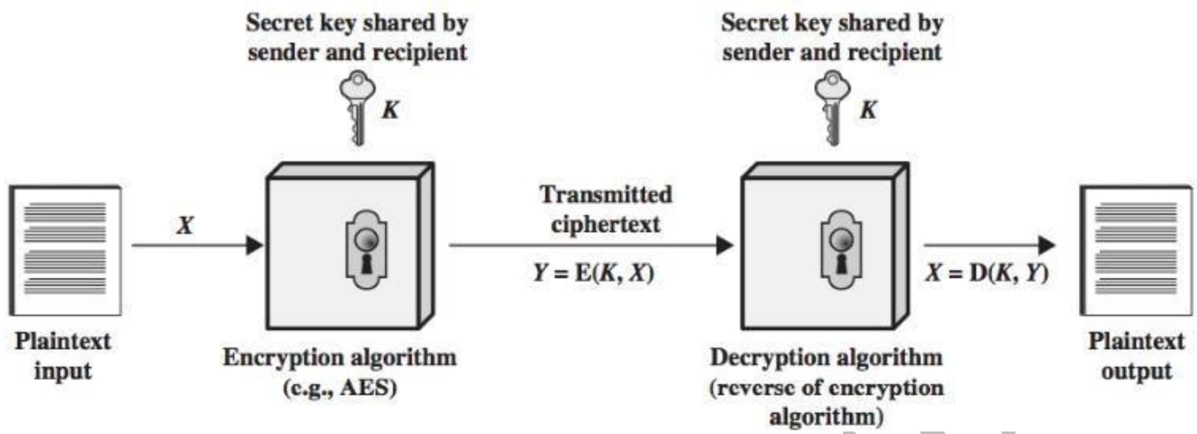


Figure2.4SimplifiedModelofSymmetricEncryption

2.7 SIMPLIFIEDDATAENCRYPTIONSTANDARD(S-DES)

TheoverallstructureofthesimplifiedDESshowninFigure2.5.TheS-DESencryption algorithm takes an 8-bit block ofplaintext (example: 10111101) and a 10-bitkeyas inputand producesan8-bitblockofciphertextas output.

The S-DES decryption algorithm takes an 8-bit block of ciphertext and the same10-bit key used to produce that ciphertext as input and produces the original 8-bit block ofplaintext.

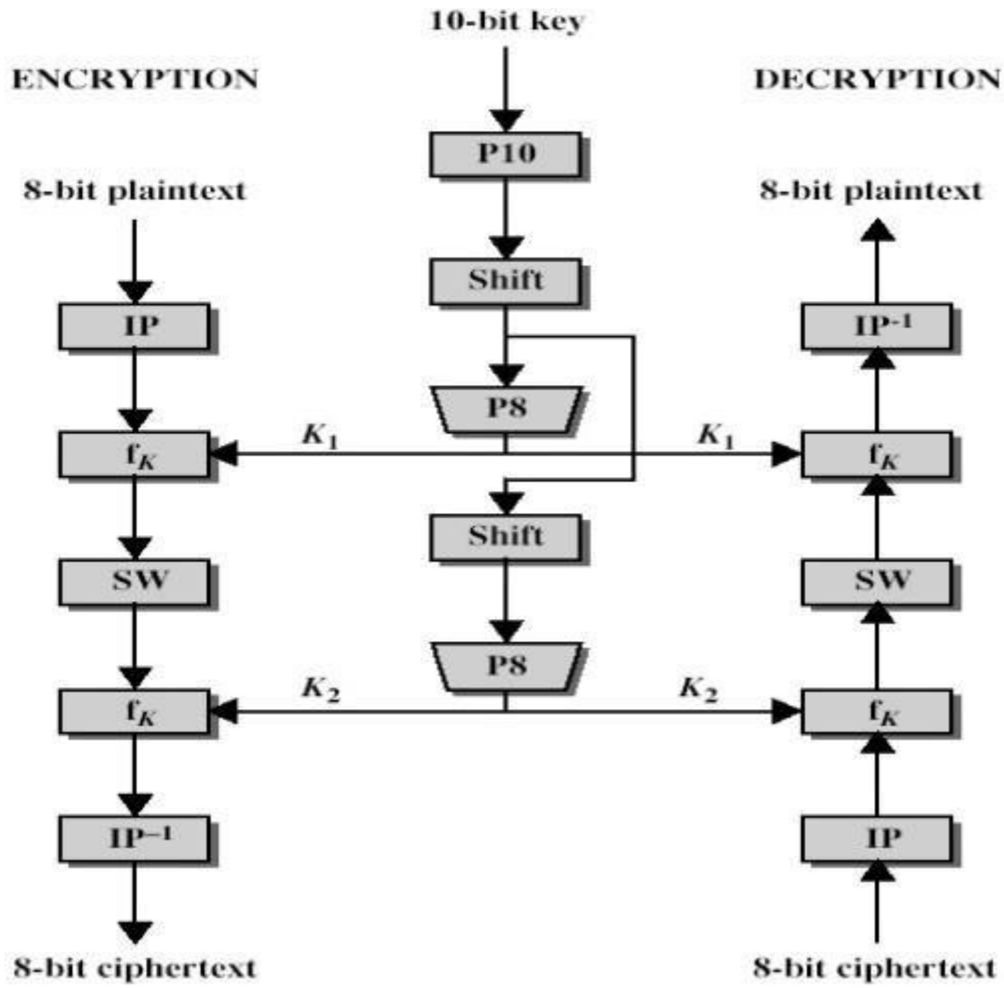


Figure2.5OverviewofS-DES Algorithm

Theencryptionalgorithminvolvesfivefunctions:

- Aninitialpermutation(IP)
- Acomplexfunctionlabeledfk,whichinvolvesbothpermutationandsubstitutiono perationsanddependson akeyinput.
- A simple permutation function that switches (SW) the two halves of thedata.
- Thefunction fkagain.

Apermutationfunctionthatisthe inverseoftheinitialpermutation

The function fk takes as input not only the data passing through the encryptionalgorithm,butalsoan8-bitkey.Herea10-bitkeyisusedfromwhichtwo8-bitsubkeysare generated.

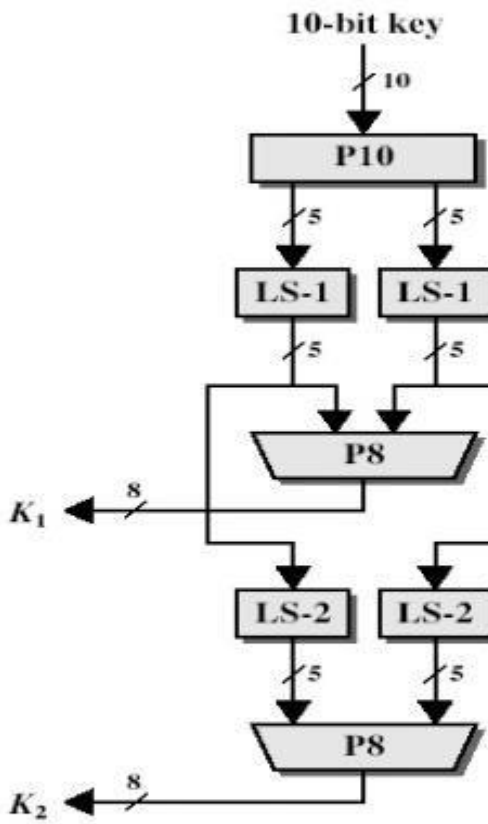
Thekeyisfirstsubjectedtoapermutation(P10).Thenashiftoperationisperformed. The output of the shift operation then passes through a permutation functionthatproducesan 8-bitoutput(P8)forthefirstsubkey (K1).

The output of the shift operation also feeds into another shift and another instanceofP8to produce thesecondsubkey (K2).

Theencryptionalgorithmcanbeexpressedasacompositioncomposition1offunctions:
IP-1ofK2 oSWofk1oIP,whichcanalsobe
writtenasCiphertext=IP-1 (fK2 (SW(fk1 (IP (plaintext)))))
Where
K1=P8(Shift(P10(Key)))
K2 =P8(Shift(shift (P10(Key))))
Decryption canbeshown asPlaintext =IP-1(fK1 (SW(fk2 (IP (ciphertext)))))

2.7.2 S-DES KeyGeneration

S-DES depends on the use of a 10-bit key shared between sender and receiver.Fromthiskey,two8-bitsubkeysareproducedforuseinparticularstagesoftheencryptionanddecryptionalgorithm.(Fi



gure2.6)

Figure2.6 S-DESKeyGeneration

First,permutethekeyinthefollowingfashion.Letthe10-bitkeybedesignatedas(k1,K2,k3,k4,k5,k6,k7,k8,k9,k10).ThenthepermutationP10 isdefinedas:

P10(k1,K2,k3,k4,k5,k6,k7,k8,k9,k10)=(k3,k5,K2,k7,k4,k1010,k1,k9,k8,k6).

P10canbeconciselydefinedbythedisplay:

P10									
3	5	2	7	4	10	1	9	8	6

This table is read from left to right; each position in the tablegives the identity oftheinputbitthatproducestheoutputbitinthatposition.So,thefirstoutputbitisbit3oftheinput;the secondoutputbitisbit5ofthe input,andsoon.

Example

The10bitkeyis(1010000010),nowfindthepermutationfromP10forthiskeyso itbecomes (1000001100).

Next, perform a circular left shift (LS-1), or rotation, separately on the first five bitsandthe secondfivebits.Inourexample,theresultis(0000111000).

Next, apply P8, which picks out and permutes 8 of the 10 bits according to thefollowingrule:

P8							
6	3	7	4	8	5	10	9

So, Theresultissubkey1(K1).In our example,thisyield (10100100).

Then go back to the pair of 5-bit strings produced by the two LS-1 functions andperforms a circular left shift of 2 bit positions on each string. In our example, the value(0000111000)becomes(0010000011).

Finally,P8isappliedagaintoproduce K2.Inourexample,theresultis(01000011).

2.7.3 S-DESEncryption

Encryptioninvolvesthesequentialapplicationoffivefunctions(Figure2.7).

1. **InitialPermutations**

The input to the algorithm is an 8-bitblockof plaintext,whichwe first permuteusingthe IP function

IP							
2	6	3	1	4	8	5	7

Theplaintextis10111101Permut
atedoutputis01111110

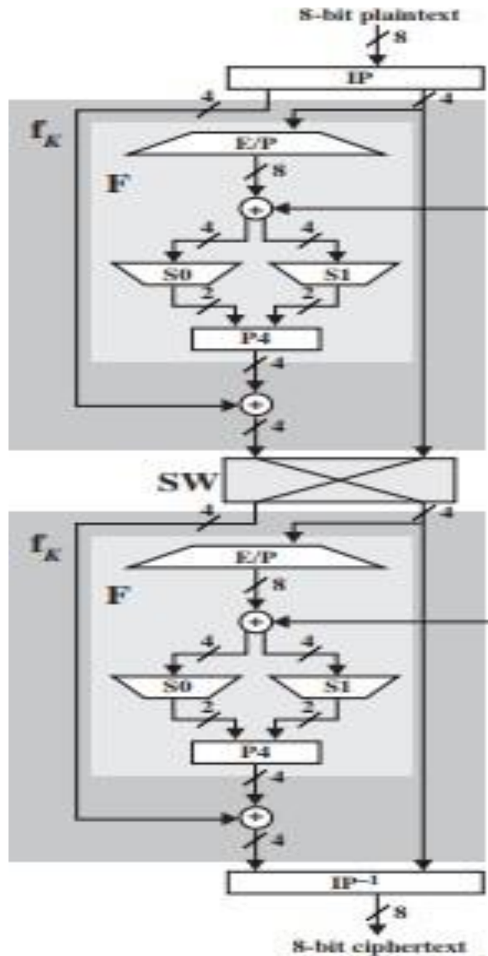


Figure2.7 S-DESEncryption

2. TheFunction f_k

The most complex component of S-DES is the function fk, which consists of acombination of permutation and substitution functions. The functions can be expressed asfollows. Let L and R be the leftmost 4 bits and rightmost 4 bits of the 8-bit input to f K, andlet F be a mapping (not necessarily one to one) from 4-bit strings to 4-bit strings. Then welet

$$Fk(L,R)=(L\oplus F\ (R,SK),R)$$

WhereSKisasub keyand⊕is the bit-by-bitexclusiveOR function

Now,describethemappingF.Theinputisa4-bitnumber(n1n2n3n4).Thefirstoperationisanexpansion/permutationoperation:

E/P							
4	1	2	3	2	3	4	1

Now,findtheE/PfromIPIP = 01111110, it becomesE/P=01111101

Now,XORwithK1

=>01111101⊕10100100=11011001

The first 4 bits (first row of the preceding matrix) are fed into the S-box S0 toproduce a 2- bit output, and the remaining 4 bits (second row) are fed into S1 to produceanother2-bitoutput.

Thesetwoboxes aredefined as follows:

$$S0 = \begin{matrix} & 0 & 1 & 2 & 3 \\ 0 & 1 & 0 & 3 & 2 \\ 1 & 3 & 2 & 1 & 0 \\ 2 & 0 & 2 & 1 & 3 \\ 3 & 3 & 1 & 3 & 2 \end{matrix} \qquad S1 = \begin{matrix} & 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 2 & 3 \\ 1 & 2 & 0 & 1 & 3 \\ 2 & 3 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 & 3 \end{matrix}$$

The S-boxes operate as follows. The first and fourth input bits are treated as a 2-bit number that specify a row of the S-box, and the second and third input bits specify acolumn of the S-box. Each s box gets 4-bit input and produce 2 bits as output. It follows00-0,01-1,10-2,11-3scheme.

Here,takefirst 4bits,

S_0=>1101

11->3

10->2 =>3=>11

Second4 bits

S1=>1001

11->3

00->0=>2=>10

So,weget1110

- Now, findP_4

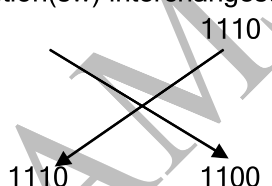
P4			
2	4	3	1

AfterP_4, thevalueis1011

Now, XORoperation1011⊕0111=>1100

3. TheSwitchfunction

- Theswitchfunction(sw) interchangestheleftandright4bits.1100



.

4. Secondfunctionf_k

- First,doE/PfunctionandXORwithK_2, thevalueis01101001⊕01000011, theanswer is00101010
- Now, findS_0andS_1

S_0=> 00->0

01->1 =>0=00

S_1=> 10->2

01->1 =>0=>00

Valueis 0000

- Now, findP_4andXORoperation

AfterP_4 => 0000 ⊕1110=1110,thenconcatenatelast4bitsafterinterchangeinsw.

- Now valueis11101100

5. FindIP-1

IP –1							
4	1	3	5	7	2	8	6

So,valueis01110101

TheCiphertextis01110101

2.8.3 S-DESDecryption

- Decryptioninvolvesthesequentialapplicationoffivefunctions.

1. **Find IP**
   - After IP,valueis 11101100
2. **Functionf_k**
   - Afterstep 2,theansweris11101100
3. **Swift**
   - Theansweris 11001110
4. **Secondf_k**
   - Theansweris01111110
5. **FindIP-1**
   - **101111101 ->Plaintext**

2.8 DATAENCRYPTIONSTANDARD

The most widely used encryption scheme is based on the Data Encryption Standard(DES) adopted in 1977. The algorithm itself is referred to as the Data Encryption Algorithm(DEA).

ForDES,dataareencryptedin64-bitblocksusinga56-bitkey.Thealgorithmtransforms 64-bitinputin aseriesofsteps intoa 64-bitoutput.

2.8.1 DESEncryption

The overall scheme for DES encryption is illustrated in the Figure 2.8. There are twoinputs to the encryption function: the **plaintext** to be encrypted and the **key**. The plaintext mustbe64bitsinlength andthe key is56bitsinlength.

2.8.2 General Depiction of DES Encryption

AlgorithmPhase 1

Looking at the left-hand side of the figure 2.8, we can see that the processing of theplaintextproceedsin threephases.

First, the 64-bit plaintext passes through an initial permutation (IP) that rearranges thebitsto producethe*permuted input*.

Phase 2:

This is followed by a phase consisting of 16 rounds of the same function, which involvesbothpermutation andsubstitutionfunctions.

The output of the last (sixteenth) round consists of 64 bits that are a function of the inputplaintext and the key. The left and right halves of the output are swapped to produce thepreoutput.

Phase 3:

Finally,thepreoutputispassedthrougha permutation(IP^{-1})thatistheinverseoftheinitialpermutation function,toproducethe64-bitciphertext. Theright-handportionofFigure showsthewayinwhichthe56-bitkeyisused.

Operationonkey:

Initially, the key is passed through a permutation function. Then, for each of the 16rounds, a *subkey* (*Ki*) is produced by the combination of a left circular shift and a permutation.Thepermutationfunctionisthesameforeachround,butadifferentsubkeyisproducedbecauseofthe repeatedshiftsofthe keybits.

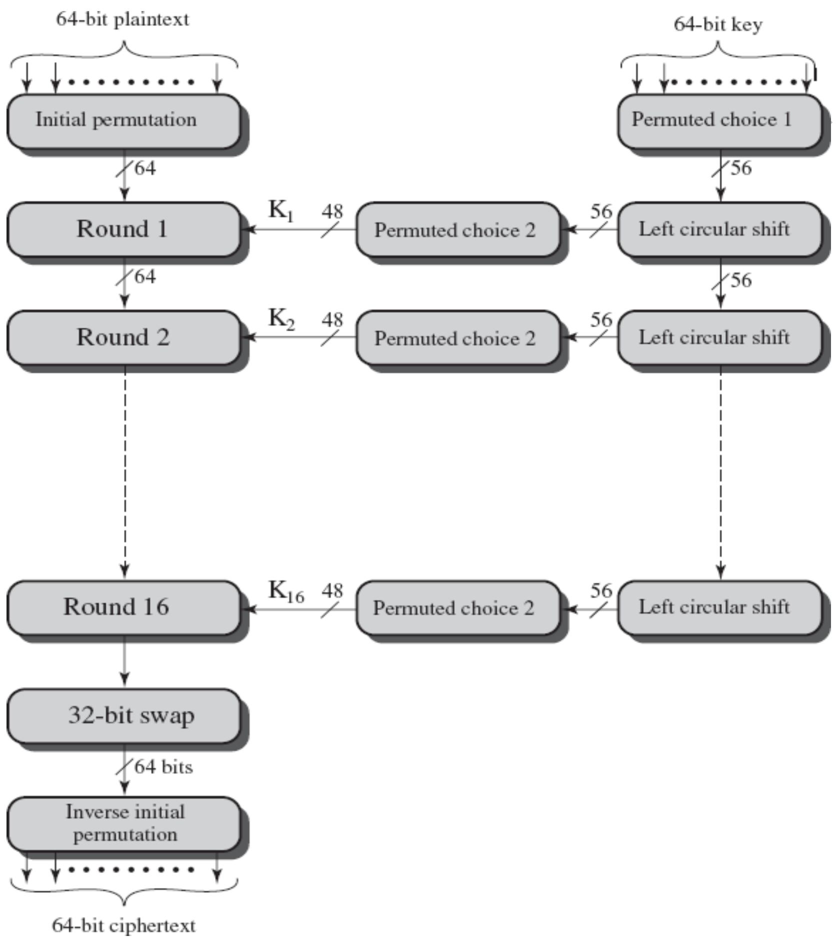


Figure2.8DESEncryption Algorithm

InitialPermutation

The input to a table consists of 64 bits numbered from 1 to 64. The 64 entries in thepermutationtablecontainapermutationofthenumbersfrom1to64.Eachentryinthepermutationtableindicatesthepositionofanumberedinputbitintheoutput,whichalsoconsistsof64 bits.

PermutationTablesforDES

(a) InitialPermutation(IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

InverseInitialPermutation(IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

ExpansionPermutation(E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

PermutationFunction(P)

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Consider thefollowing64-bitinput*M*:

*M*1	*M*2	*M*3	*M*4	*M*5	*M*6	*M*7	*M*8
*M*9	*M*10	*M*11	*M*12	*M*13	*M*14	*M*15	*M*16
*M*17	*M*18	*M*19	*M*20	*M*21	*M*22	*M*23	*M*24
*M*25	*M*26	*M*27	*M*28	*M*29	*M*30	*M*31	*M*32
*M*33	*M*34	*M*35	*M*36	*M*37	*M*38	*M*39	*M*40
*M*41	*M*42	*M*43	*M*44	*M*45	*M*46	*M*47	*M*48
*M*49	*M*50	*M*51	*M*52	*M*53	*M*54	*M*55	*M*56
*M*57	*M*58	*M*59	*M*60	*M*61	*M*62	*M*63	*M*64

where*Mi*isa binarydigit. Thenthepermutation*X*= IP(*M*) isasfollows:

*M*58	*M*50	*M*42	*M*34	*M*26	*M*18	*M*10	*M*2
*M*60	*M*52	*M*44	*M*36	*M*28	*M*20	*M*12	*M*4
*M*62	*M*54	*M*46	*M*38	*M*30	*M*22	*M*14	*M*6
*M*64	*M*56	*M*48	*M*40	*M*32	*M*24	*M*16	*M*8
*M*57	*M*49	*M*41	*M*33	*M*25	*M*17	*M*9	*M*1
*M*59	*M*51	*M*43	*M*35	*M*27	*M*19	*M*11	*M*3
*M*61	*M*53	*M*45	*M*37	*M*29	*M*21	*M*13	*M*5
*M*63	*M*55	*M*47	*M*39	*M*31	*M*23	*M*15	*M*7

Inverse permutation $Y = IP^{-1}(X) = IP^{-1}(IP(M))$,Therefore we can see that the original ordering ofthebitsisrestored.

2.8.3 DetailsofSingleRound

The below figure 2.9 shows the internal structure of a single round. The left and right halves ofeach 64-bit intermediate value are treated as separate 32-bit quantities, labeled L (left) and R(right).Theoverallprocessingateach round canbesummarizedin the followingformulas:

$L_i=R_{i-1}$

$R_i=L_{i-1} x F(R_{i-1},K_i)$

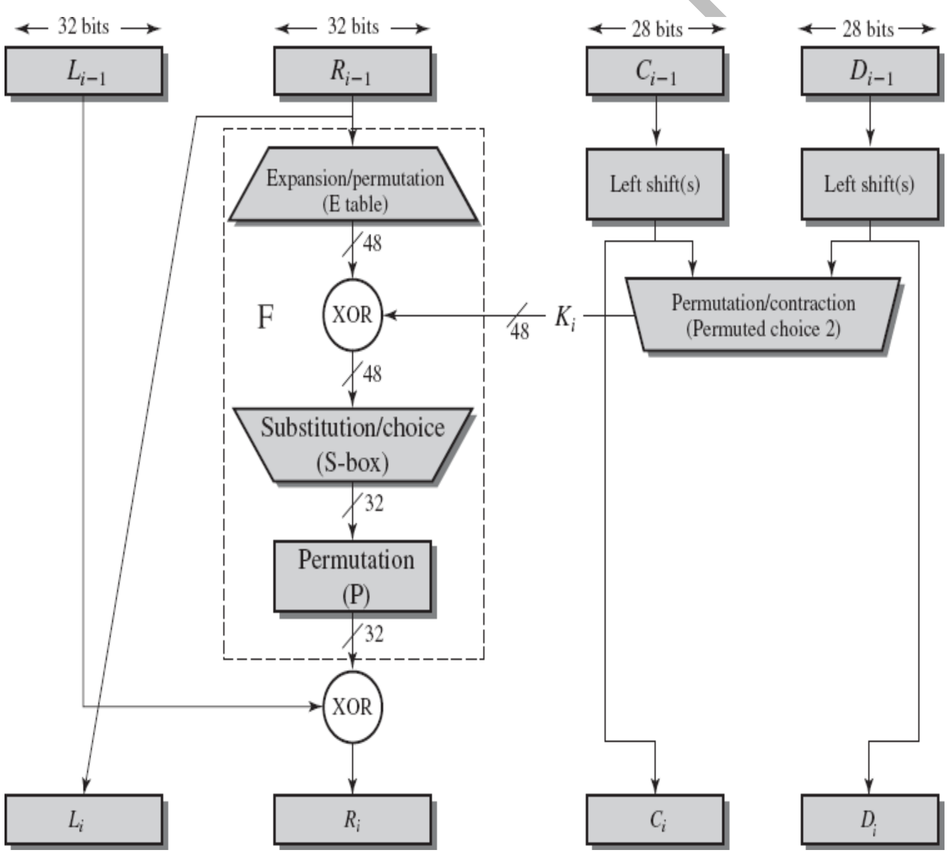


Figure2.9SingleRoundofDES Algorithm

The round key K_i is 48 bits. The *R* input is 32 bits. This *R* input is first expanded to 48 bits byusing a table that defines a permutation plus an expansion that involves duplication of 16 of the*R* bits. The resulting 48 bits are XORed with K_i. This 48-bit result passes through a substitutionfunctionthatproducesa32-bitoutput,whichisthenpermuted.

DefinitionofS-Boxes

The substitution consists of a set of eight S-boxes, each of which accepts 6 bits as inputand produces 4 bits as output. The first and last bits of the input to box S*i* form a 2-bit binarynumber to select one of four substitutions defined by the four rows in the table for S*i*. The middlefourbitsselectoneofthesixteencolumnsas showninfigure2.10.

Thedecimalvalueinthecellselectedbytherowandcolumnisthenconvertedtoits4-bitrepresentation toproducethe output.

For example, in S1 for input 011001, the row is 01 (row 1) and the column is 1100(column12).The valueinrow 1,column12is9,sotheoutputis1001.

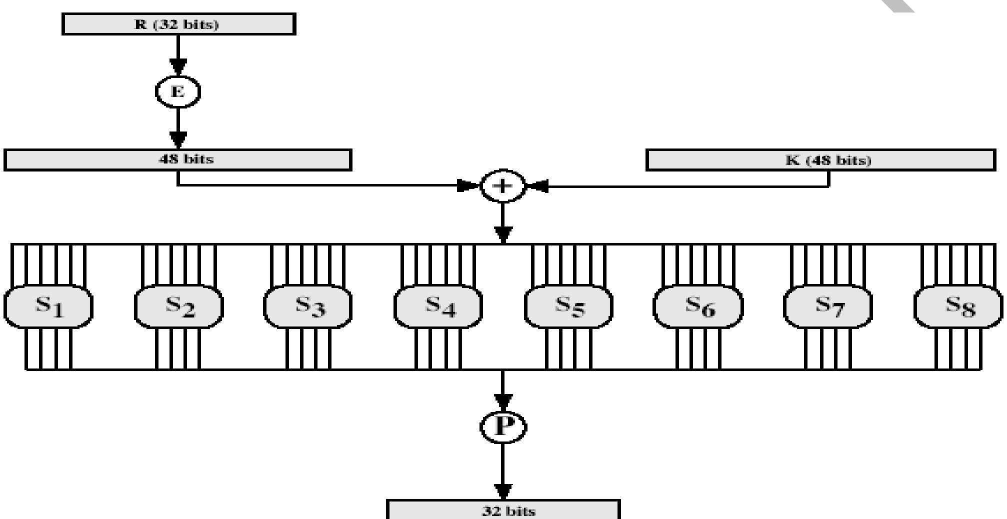


Fig2.10 CalculationofF(R, K)

2.8.4 KeyGeneration

The 64-bit key is used as input to the algorithm. The bits of the key are numbered from 1through 64; every eighth bit is ignored. The key is first subjected to a permutation governed by atable labeled Permuted Choice One. The resulting 56-bit key is then treated as two 28-bitquantities,labeledC_0andD_0.

Ateachround,C_{i-1}andD_{i-1}areseparatelysubjectedtoacircularleftshift,orrotation,of 1 or 2 bits. These shifted values serve as input to the next round. They also serve as input toPermuted Choice 2, which produces a 48-bit output that serves as input to the function F(R_{i-1},K_i).

DESKeySchedule Calculation

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

(b) Permuted ChoiceOne(PC-1)

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

(c) PermutedChoiceTwo(PC-2)

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

(d) Schedule ofLeft Shifts

Roundnumber:12 345678 9101112 13141516

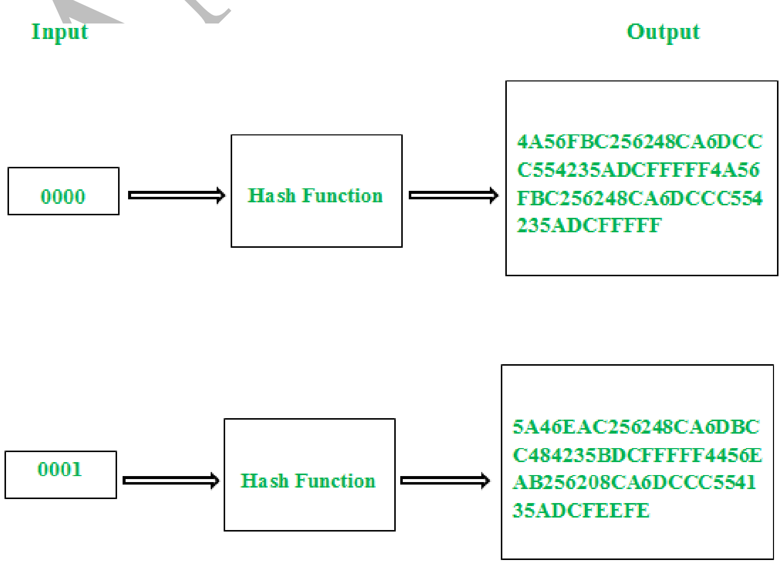
Bitsrotated:11222222 12 2 2 2 2 2 1

2.8.5 DESDecryption:

As with any Feistel cipher, decryption uses the same algorithm as encryption,except that theapplicationofthesubkeysisreversed.Additionally,theinitialandfinalpermutationsarereversed.

2.8.6 TheAvalancheEffect:

A desirable property of any encryption algorithm is that a small change in either the plaintext orthekeyshouldproduceasignificantchangeintheciphertext.Inparticular,achangeinonebitoftheplain textoronebitofthekey shouldproduceachange inmanybitsoftheciphertext.



2.9 THESTRENGTHOFDES

Thestrength ofDESdependsontwofactors:**keysize**andthe**natureofthealgorithm**.

1. **TheUseof56-BitKeys**
   With a key length of 56 bits, there are 2^{56} possible keys, which is approximately 7.2 x10^{16}.Thus,a brute-force attackappearsimpractical.

2. **TheNatureoftheDESAlgorithm**
   In DES algorithm, eight substitution boxes called S-boxes that are used in each iteration.Because the design criteria for these boxes, and indeed for the entire algorithm, were not madepublic, there is a suspicion that the boxes were constructed in such a way that cryptanalysis ispossibleforanopponentwhoknowstheweaknessesintheS-boxes.Despitethis,noonehasso farsucceededin discoveringthe supposedfatalweaknessesinthe S-boxes.

3. **TimingAttacks**
   A timing attack is one in which information about the key or the plaintext is obtained byobservinghowlongittakesagivenimplementationtoperformdecryptionsonvariousciphertexts. A timing attack exploits the fact that an encryption or decryption algorithm oftentakes slightly differentamountsoftimeon differentinputs.

2.9.1 AttacksonDES:

Twoapproachesare:

1. Differentialcryptanalysis
2. Linearcryptanalysis

2.9.1.1 DifferentialCryptanalysis

Differential cryptanalysis is the first published attack that is capable of breaking DES in lessthan255complexities.TheneedtostrengthenDESagainstattacksusingdifferentialcryptanalysisplayedalargepartinthedesignoftheS-boxesand thepermutationP.

- One ofthemostsignificant recent(public)advancesincryptanalysis
- Powerfulmethodtoanalyzeblockciphers
- Used toanalyzemostcurrentblock cipherswithvaryingdegrees ofsuccess

DifferentialCryptanalysisAttack:

Thedifferentialcryptanalysisattackiscomplex.Therationalebehinddifferentialcryptanalysis is to observe the behavior of pairs of text blocks evolving along each round of thecipher,insteadofobservingthe evolutionofasingletextblock.

Consider the original plaintext block *m* to consist of two halves *m*0, *m*1. Each round ofDES maps the right-hand input into the left-hand output and sets the right-hand output to be afunctionofthe left-handinputand thesubkeyforthisround.

So,at eachround,only onenew 32-bitblockiscreated. Ifwe labeleach newblock m_1(2≤ i≤17),thentheintermediatemessagehalvesarerelatedasfollows:

$$m_{i+1}=m_{i-1}\oplus f(m_i,K_i),i=1,2,...,16$$

Indifferentialcryptanalysis,westartwithtwomessages,*m*and*m*',withaknownXORdifference $\Delta m= m \oplus m'$, and consider the difference between the intermediate message halves: $m_i=m_i\oplus m_i'$ Thenwehave:

$$\begin{aligned}\Delta m_{i+1} &= m_{i+1} \oplus m"_{i-1}\\ &=[m_{i-1} \oplus f(m_i,k_i])] \oplus [m"_{i-1}\oplus f(m"_i,k_i)]\\ &=\Delta m_{i-1} \oplus [f(m_i,k_i) \oplus f(m"_i,ki)]\end{aligned}$$

Let us suppose that there are many pairs of inputs to f with the same difference yield thesameoutputdifference ifthe samesubkey isused.

Therefore, if we know Δm_{i-1} and $\Delta m_i$with high probability, then we know Δm_{i+1} with highprobability.Furthermore,ifanumberofsuchdifferencesaredetermined,itisfeasibletodetermineth e subkey usedinthefunction f.

2.9.1.2 LinearCryptanalysis

Thisattackis basedonthefactthatlinearequationcanbeframed todescribethetransformations. TheprincipleoflinearcryptanalysisisasfollowsLengthofC

TandPT=nbits;
key=mbit
Blockofciphertextisc[1]c[2]…c[n];Blockof
keyisk[1]k[2]….k[m]

A[I,j,..k]=A[i]⊕A[j]⊕.⊕A[k]

- CanattackDESwith247knownplaintexts, stillinpracticeinfeasible
- Find linearapproximations withprobp!= ½
- $P[i_1,i_2,...,i_a](+)c[j_1,j_2,...,j_b]=k[k_1,k_2,...,k_c]$Where$i_a,j_b,k_c$arebit locationsinp,c,k

2.10 BLOCKCIPHERPRINCIPLES

Therearethreecriticalaspectsofblockcipherdesign:

1. Numberofrounds,
2. Design ofthefunction F
3. Keyscheduling.

NumberofRounds

- When the greater the number of rounds, the more difficult it is to perform cryptanalysis,evenfora relativelyweakF.
- Thenumberof rounds ischosensothatknown cryptanalyticeffortsrequire greatereffortthanasimplebrute-forcekey searchattack
- WhenroundDESS=16,adifferentialcryptanalysisattackisslightlylessefficientthanbruteforce ,the differentialcryptanalysisattackrequires 2^{55}operations.
- Itmakesiteasytojudgethestrengthofanalgorithmand to compare differentalgorithms.

DesignofFunctionF

Thisisthemost important function

Criterianeededfor F,

- Itmust bedifficultto“unscramble”thesubstitutionperformed byF.
- The function should satisfy **strict avalanche criterion (SAC)** which statesthatanyoutput bit *j* of an S-box should change with probability 1/2 when any single input bit *i* isinverted forall*i*,*j*.
- The function should satisfy **bit independence criterion(BIC)**, which states that outputbits *j* and *k* should change independently when any single input bit *i* is inverted for all *i*, *j*,and*k*.

KeyScheduleAlgorithm

- Thekeyisusedtogenerateonesubkeyforeachround.
- The sub keys to maximize the difficulty of deducing individual sub keys and the difficultyofworkingbackto the mainkey.

2.10.1 StreamCipherandBlockCipher

Astreamcipherisonethatencryptsadigital data streamone bitoronebyteatatime.E.g,vigenerecipher.Figure(2.11a)

Ablockcipher
isoneinwhichablockofplaintextistreatedasawholeandusedtoproduceaciphertextblockofequallength. Typically,a blocksize of64or128 bitsisused. Figure(2.11b)

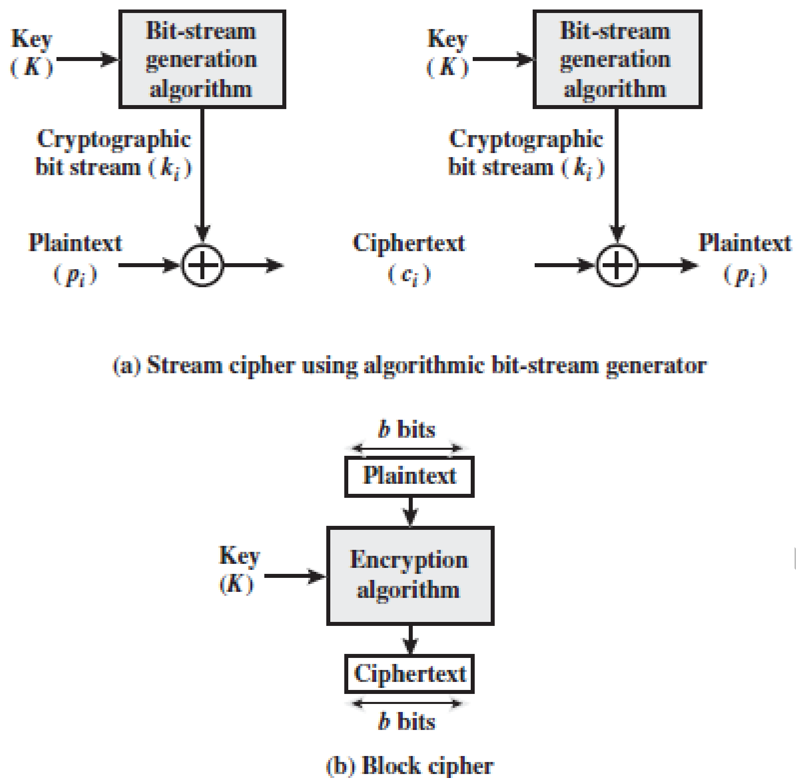
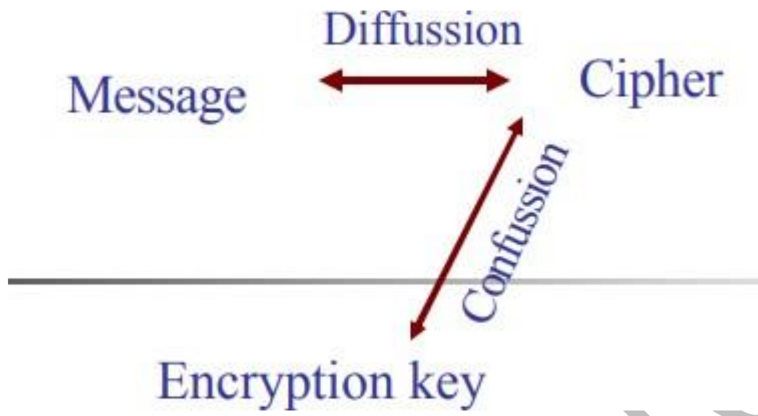


Figure2.11StreamCipher andBlockCipher

- ManyblockciphershaveaFeistelstructure.Suchastructureconsistsofanumberofidenticalrou ndsofprocessing.
- Ineachround,asubstitutionisperformedononehalfofthedatabeingprocessed,followedbyaper mutationthatinterchangesthe twohalves.
- Theoriginalkeyisexpandedsothatadifferentkey isusedforeach round.
- TheDataEncryptionStandard(DES)hasbeenthemostwidelyusedencryptionalgorithm.Itexhi bitstheclassicFeistelstructure.
- The DES uses a 64-bit block and a 56-bit key. Two important methods of cryptanalysisare differential cryptanalysis and linear cryptanalysis. DES has been shown to be highlyresistanttothesetwotypesofattack.
- A block cipher operates on a plaintext block of n bits to produce a ciphertext block of nbits. There are possible different plaintext blocks and, for the encryption to be reversible(i.e., for decryption to be possible), each must produce a unique ciphertext block. Such atransformationiscalledreversible,ornonsingular
- Inparticular,Feistelproposedtheuseofacipherthatalternatessubstitutionsandpermutations, wherethesetermsaredefinedasfollows:
  - **Substitution:** Each plaintext element or group of elements is uniquely replaced byacorrespondingciphertextelementorgroupofelements.
  - **Permutation:** A sequence of plaintext elements is replaced by a permutation ofthatsequence.Thatis,noelementsareaddedordeletedorreplacedinthesequence,rat hertheorderinwhichtheelementsappearinthesequenceischanged.

- Twomethodsforfrustratingstatisticalcryptanalysis are:
  - **Diffusion –** Each plaintext digit affects many ciphertext digits, or each ciphertextdigitisaffectedby manyplaintextdigits.
  - **Confusion** – Makethestatisticalrelationshipbetweenaplaintextandthecorresponding ciphertext as complex as possible in order to thread attempts todeducethekey.



2.10.2 Feistelcipher structure

- Theleft-hand sideof figure 2.12depictsthestructureproposed byFeistel.
- The input to the encryption algorithm is a plaintext block of length 2w bits and a key K. theplaintextblockisdividedintotwohalvesL_0andR_0.
- The two halves of the data pass through n rounds of processing and then combine toproducetheciphertextblock.Eachroundi hasinputsL_{i-1} andR_{i-1},derivedfromthepreviousround,aswellasthe subkey K_i,derived from the overallkey K.
- In general, the subkeys K_i are different from K and from each other. All rounds have thesamestructure.
- A substitution is performed on the left half of the data (as similar to S-DES). This is doneby applying a round function F to the right half of the data and then taking the XOR of theoutputofthatfunctionandthe lefthalfofthe data.
- Theroundfunctionhasthesamegeneralstructureforeachroundbutisparameterizedbytheroundsubkeyk_i.Followingthissubstitution,apermutationisperformedthatconsistsofthe interchangeofthe twohalvesofthedata.
- Thisstructureisaparticular formofthesubstitution-permutationnetwork.

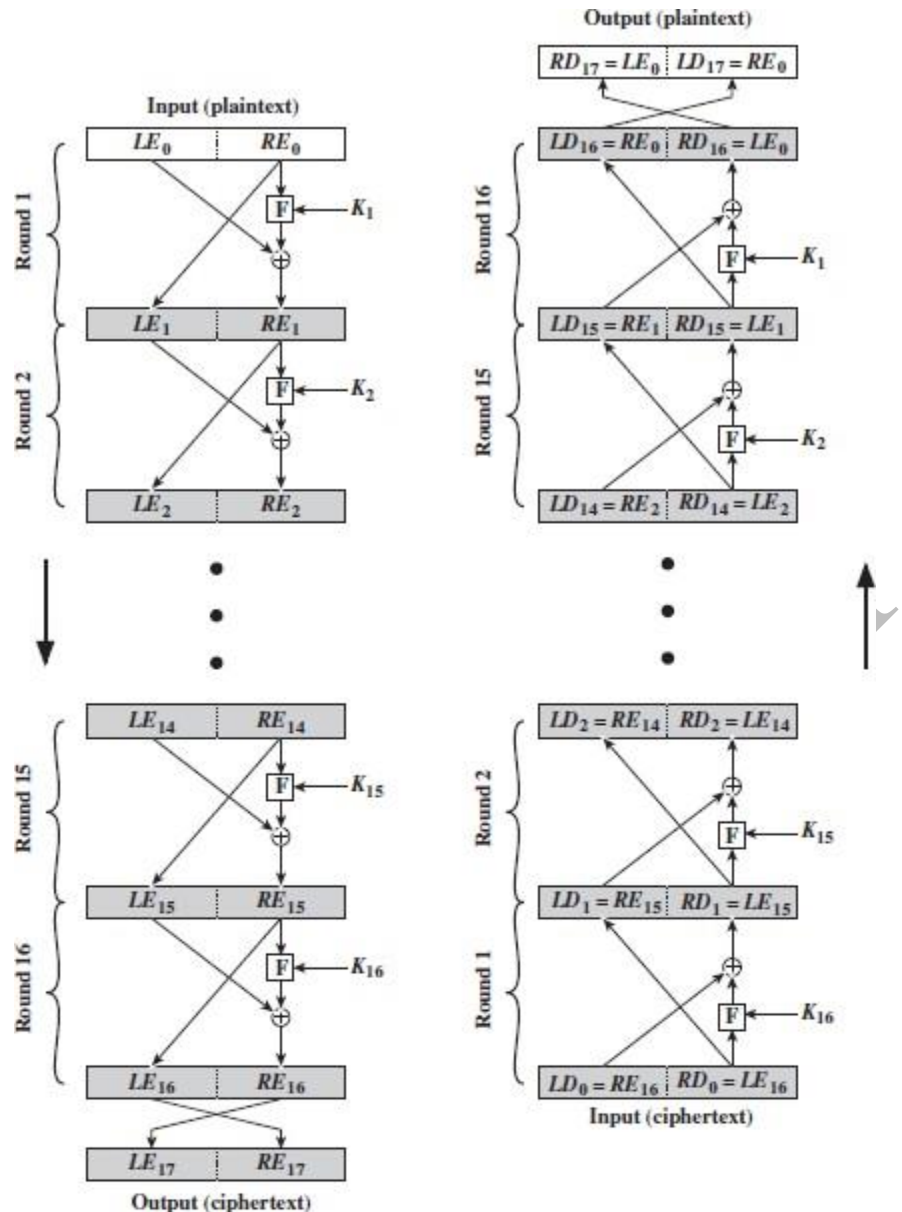


Figure2.12FeistelEncryptionand Decryption(16rounds)

Thefeatures ofFeistelnetworkare:

- **Blocksize**-Increasingsizeimproves security,butslows cipher
- **Keysize**-Increasingsizeimprovessecurity,makesexhaustivekeysearchingharder,butmay slow cipher
- **Numberofrounds**-Increasing number improves security,butslows cipher
- **Subkeygeneration**- Greatercomplexitycanmakeanalysisharder,butslowscipher
- **Roundfunction**- Greatercomplexitycanmakeanalysisharder,but slowscipher

- Theprocessofdecryption is essentially thesameas theencryption process.
- Theruleisasfollows:usetheciphertextasinputtothealgorithm,butusethesubkeyk_i inreverseorder.i.e., $k_n$inthe firstround,k_{n-1}insecondround and soon.
- For clarity, we use the notation LE_i and RE_i for data traveling through the decryptionalgorithmand$LD_i$and RD_i.
- The above diagram indicates that, at each round, the intermediate value of the decryptionprocess is same (equal) to the corresponding value of the encryption process with twohalvesofthe valueswapped.

i.e., RE_i || LE_i (or) equivalently RD_{16-i} || LD_{16-i}

- Afterthelastiterationoftheencryptionprocess,thetwohalvesoftheoutputareswapped,sothatthe ciphertextis RE_{16}|| LE_{16}.
- Theoutputofthatroundistheciphertext.Nowtaketheciphertextanduseitasinputtothesamealgorithm.
- TheinputtothefirstroundisRE$_{16}$||LE$_{16}$,whichisequaltothe32-bitswapoftheoutputofthesixteenthround oftheencryptionprocess.
- Nowwewillseehowtheoutputofthefirstroundofthedecryptionprocessisequaltoa32-bitswapofthe inputtothesixteenthroundoftheencryptionprocess.
- First consider theencryption process,

$$\mathbf{LE_{16}=RE_{15}}$$

$$\mathbf{RE_{16}=LE_{15}\oplus F(RE_{15},K_{16})}$$

Onthedecryptionside,

$$LD_1=RD_0=LE_{16}=RE_{15}RD_1$$
$$=LD_0\oplus F(RD_O,K_{16})$$
$$=RE_{16}\oplus F(RE_{15},K_{16})$$
$$= [LE_{15}\oplus F(RE_{15},K_{16})]\oplus F(RE_{15},K_{16})$$
$$=LE_{15}$$

Therefore, $LD_1=RE_{15}, RD_1=LE_{15}$

Ingeneral, fortheithiteration oftheencryption algorithm,

$$\mathbf{LE_i=RE_{i-1}}$$

$$\mathbf{RE_i=LE_{i-1}\oplus F(RE_{i-1},K_i)}$$

- Finally,theoutputofthelastroundofthedecryptionprocessisRE$_0$||LE$_0$.A32-bitswaprecoverstheoriginalplaintext.

2.11 BLOCKCIPHERMODESOFOPERATION

- BlockCipheristhe basicbuildingblock toprovide datasecurity.
- Toapplytheblockciphertovariousapplications,NISThasproposed4modesofoperation. Theblockcipherisusedtoenhancethesecurity oftheencryptionalgorithm

2.11.1 MultipleEncryptionandTripleDES

Thevulnerability ofDEStoabrute-force attackhasbeendetectedbyusingtwo approaches areshowninfigure2.13

1. One approachis todesign a completelynewalgorithm,ofwhich AES isa primeexample
2. Another alternative, which would preserve the existing investment in software andequipment,is tousemultipleencryptionswithDESand multiplekeys.

DoubleDES

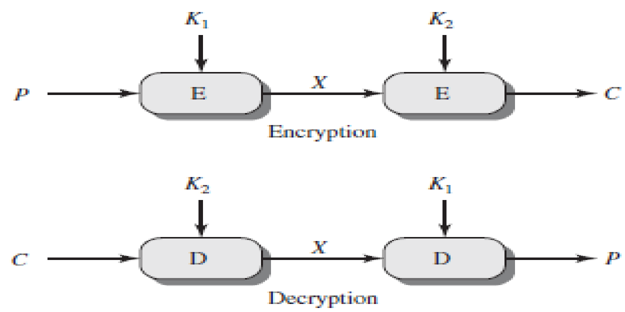
Thesimplestformof multipleencryptions hastwoencryptionstagesand two keys.GivenaplaintextPandtwoencryptionkeysK_1andK_2,ciphertextCis generated as

$$C = \mathrm{E}(K_2, \mathrm{E}(K_1, P))$$

Decryption requiresthat thekeysbeappliedinreverseorder:

$$P = \mathrm{D}(K_1, \mathrm{D}(K_2, C))$$

ForDES, thisschemeapparentlyinvolves akeylengthof56*2 =112bits,resultingin adramaticincreaseincryptographic strength.



(a) Double encryption

Figure2.13MultipleEncryption

Reductionto aSingleStage

SupposeitweretrueforDES,forall56-bitkeyvalues,thatgivenanytwokeysK_1andK_2,itwouldbepossibletofindakey K_3suchthat

$$E(K_2, E(K_1, P)) = E(K_3, P)$$

Meet-in-the-MiddleAttack

The use of double DES results in a mapping that is not equivalent to a single DESencryption. But there is a way to attack this scheme, one that does not depend on any particularproperty of DES but that will work against any block encryption cipher. This algorithm, known asameet-in-the-middleattack.

It is based on the observation that, if we

$$C = E(K_2, E(K_1, P))$$

haveThen

$$X = E(K_1, P) = D(K_2, C)$$

Given a known pair, (P, C), the attack proceeds as follows.First, encrypt P for all 256possiblevaluesofK_1.StoretheseresultsinatableandthensortthetablebytheValuesofX.

Next,decryptCusingall256possiblevaluesofK2.Aseachdecryptionisproduced, checktheresultagainstthetable for a match.

If a match occurs, then test the two resulting keys against a new known plaintext-ciphertextpair.Ifthetwokeysproducethecorrectciphertext,acceptthemasthecorrectkeys.

ForanygivenplaintextP,thereare264possibleciphertextvaluesthatcouldbeproduced by double DES. Double DES uses, in effect, a 112-bit key, so that there are 2112possiblekeys.

TripleDESwithTwoKeys

To overcome the meet-in-the-middle attack is to use three stages of encryption with threedifferentkeys. ThisiscalledadTripleDESor3DESasshowninfigure2.14.

Theknownplaintextattackin2^{112}.Thekeylength of56* 3=168bits whichisadrawback.

Tuchmanproposeda triple encryption methodthatusesonlytwokeysgivenplaintext k_1,k_2 .Thefinalcipher textis

$$C = E(K_1, D(K_2, E(K_1, P)))$$
$$P = D(K_1, E(K_2, D(K_1, C)))$$

- Thefunctionfollowsanencrypt-decrypt-encrypt(EDE)sequence

Itsonly advantage isthat itallowsusersof3DEStodecryptdataencryptedbyusersoftheoldersingleDES:

$$C = E(K_1, D(K_1, E(K_1, P))) = E(K_1, P)$$
$$P = D(K_1, E(K_1, D(K_1, C))) = D(K_1, C)$$

- 3DESwithtwokeys isarelativelypopularalternativetoDES
- There are nopractical cryptanalyticattackson3DES.
- Thecostofabrute-forcekey search on3DESisontheorderof2^{112}

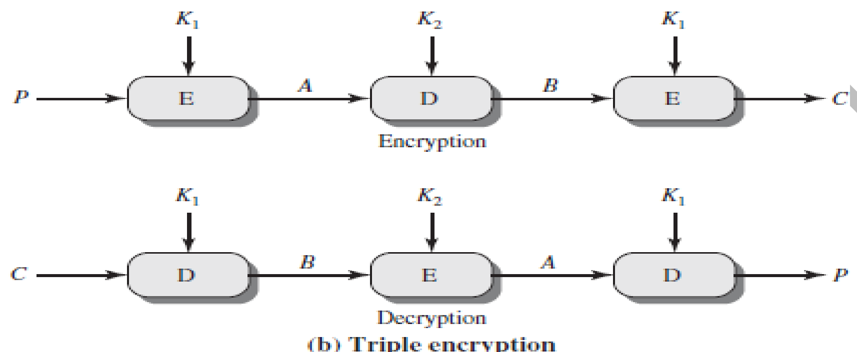


Figure2.14TripleDES

Thefirstseriousproposalcamefrom MerkleandHellman

1. MerkleandHellman

Theconceptisto findplaintextvaluesthat producea first intermediatevalueofA= 0and thenusingthe meet-in-the-middleattack todeterminethetwokeys.

- Thelevel ofeffortis2^{56},
- Thetechniquerequires256chosenplaintext-ciphertextpairs,whichisanumberunlikely tobeprovided.

2. known-plaintextattack:

The attack is based on the observation that if we know A and Cthen the problemreduces tothatofan attackondouble DES.

The attacker does not know A, even if P and C are known, as long as the two keys areunknown. The attacker can choose a potential value of A and then try to find a known (P, C) pairthatproducesA.

Theattackproceeds asfollows.

Step 1:

- Obtainn(P,C)pairs.Thisistheknownplaintext.PlacetheseinatablesortedonthevaluesofP

Step 2:

- PickanarbitraryvalueaforA,andcreateasecondtablewithentriesdefinedinthefollowing fashion.
- Foreachofthe2^{56}possible keysK_1=i, calculatetheplaintextvaluePi thatproducesa.
- Foreach$P_i$thatmatchesanentryinTable1,createanentryinTable2consistingoftheK_1valueand thevalueofBthatisproduced.

Step 3:

- WenowhaveanumberofcandidatevaluesofK$_1$inTable2andareinapositiontosearch foravalueofK$_2$.
- Foreachofthe256possiblekeysK$_2$=j,calculatethesecondintermediatevalueforourchosenval ueofa
- Ifthereis a match,thenthecorrespondingkeyifromTable2 plus thisvalueofjarecandidatevaluesforthe unknownkeys(K1,K2).

Step 4:

- Testeachcandidatepairofkeys(i,j)onafewotherplaintext-ciphertextpairs.
- Ifa pairof keysproducesthedesiredciphertext, thetaskiscomplete.Ifno pairsucceeds,repeatfromstep1withanew valueofa.

2.11.2 MODE1:ElectronicCodeBook

The simplest mode is the electronic codebook (ECB) mode shown in figure2.15. Hereplaintext is handled one block at a time and each block of plaintext is encrypted using the samekey.

The term codebook is used because, for a given key, there isa unique cipher text for everyb-bitblockofplaintext.

When the message longer than b bits, to break the message into b-bit blocks.For the lastblockwhentheno ofbitsisless thanb,paddingthelastblockifnecessary.

Decryption is performedoneblockatatime, alwaysusingthesamekey.

Uses: TheECBmethodisidealforashortamountofdata,suchasan encryptionkey.

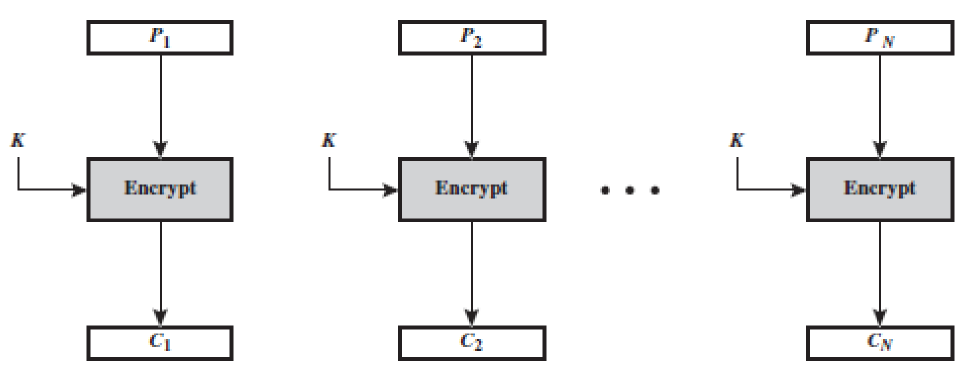
Disadvantage:

Whenb‟ -bitblockofplaintextappearsmorethanonce inthemessage,it alwaysproduces the sameciphertextoutput.

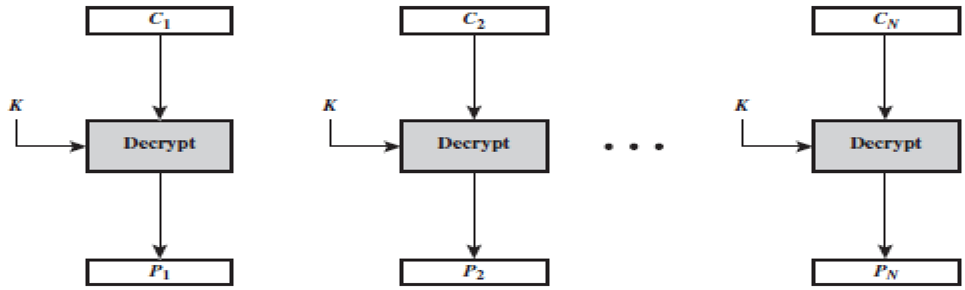
Forlengthymessages, theECBmodemaynotbe secure.If themessageishighlystructured,itmaybepossible for acryptanalysttoexploitthese regularities.

If the message has repetitive elements with a period of repetition a multiple of b bits, thentheseelementscan beidentifiedby theanalyst.

Thismayhelp intheanalysisormayprovideanopportunityfor substitutingorrearrangingblocks.



(a) Encryption



(b) Decryption

Figure2.15ElectronicCodeBook(ECB)Mod

ePropertiesforEvaluatingandConstructing ECB

Overhead: Theadditionaloperationsfortheencryptionanddecryptionoperationwhencompared to encryptinganddecryptingin theECBmode.

Error recovery: The property that an error in the *i*th cipher text block is inherited by only a fewplaintextblocks

Error propagation: It is meant here is a bit error that occurs in the transmission of a cipher textblock, not a computational error in the encryption of a plaintext block.**Diffusion:**Lowentropyplaintextblocksshouldnotbereflectedintheciphertextblocks.Roughly,low entropy equatestopredictabilityorlackofrandomness

Security: Whetherornot the ciphertextblocksleak informationabouttheplaintextblocks.

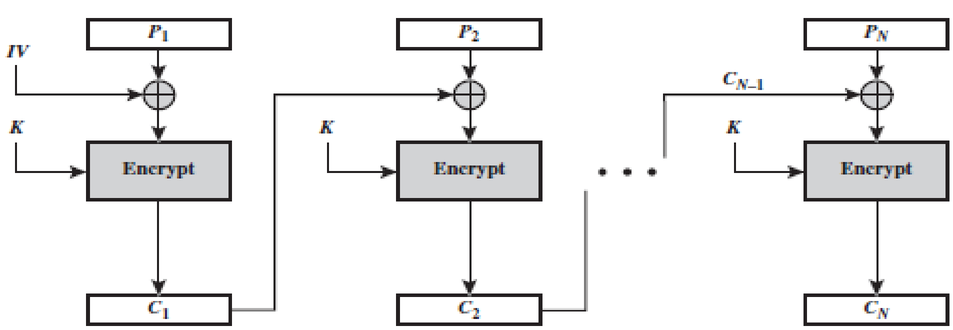
2.11.3 MODE2:CipherBlockChainingMode

ThismethodistoovercomethedisadvantageofECB(i.e)whenthePTblockisrepeatedCBCproducesdifferentciphertextblocks

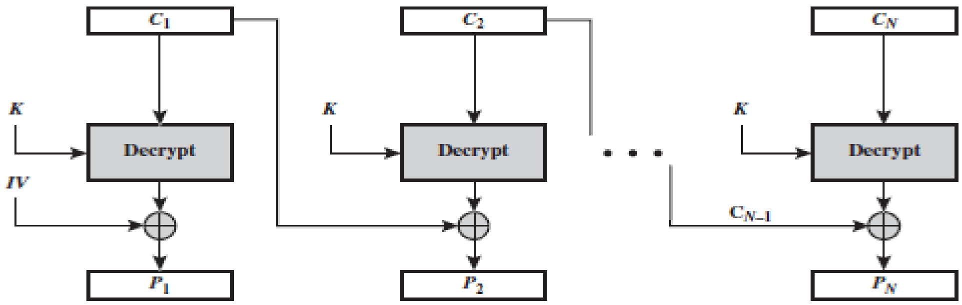
The input to the encryption function for each plaintext block bears no fixed relationship totheplaintextblock.Therefore,repeatingpatternsofbbitsarenotexposed.

For decryption, each cipher block is passed through the decryption algorithm. The resultis XORed with the preceding cipher text block to produce the plaintext block are shown in figure2.16.

$$C_j = E(K, [C_{j-1} \oplus P_j])$$



(a) Encryption



(b) Decryption

Figure2.16CipherBlockChaining(CBC)Mode

Then

$$D(K, C_j) = D(K, E(K, [C_{j-1} \oplus P_j]))$$

Toproducethefirst blockofciphertext, aninitialization vector(IV) isXORedwith thefirstblockofplaintext.

On decryption, theIV isXORedwith theoutputof thedecryptionalgorithm to recover thefirstblockofplaintext.

SizeofIV=Size ofdataBlocksWecan defineCBC modeas

CBC	$C_1 = E(K, [P_1 \oplus IV])$ $C_j = E(K, [P_j \oplus C_{j-1}])\ j = 2, \ldots, N$	$P_1 = D(K, C_1) \oplus IV$ $P_j = D(K, C_j) \oplus C_{j-1}\ j = 2, \ldots, N$

Formaximumsecurity,theIVshouldbeprotectedagainstunauthorizedchanges. Thiscouldbedone by sendingthe IVusingECBencryption

Reasonforprotectingthe IV:

Ifanopponentis able tofool thereceiverintousinga differentvalueforIV,thentheopponentisabletoinvertselected bitsinthefirstblockofplaintext.Toseethis,consider

$$C_1 = E(K, [IV \oplus P_1])$$
$$P_1 = IV \oplus D(K, C_1)$$

Nowusethenotation thatX[i]denotes theithbitof theb-bitquantityX.Then

$$P_1[i] = IV[i] \oplus D(K, C_1)[i]$$

Then,usingthepropertiesofXOR,we canstate

$$P_1[i]' = IV[i]' \oplus D(K, C_1)[i]$$

Where the prime notation denotes bit complementation. This means that if an opponentcan predictably change bits in IV, the corresponding bits of the received value of P1 can bechanged.

2.11.4 MODE3: CipherFeedback Mode:

We know that the DES is a block cipher.it is possible to convert block cipher into stream CipherusingCFBmode

TheadvantagesofCFBisthat

- Eliminatestheneedtopadamessage
- It alsocan operateinrealtime
- Thelengthof theCT =Length ofPT

Figure2.17depictstheCFBscheme.Inthefigure2.17,itisassumedthattheunitoftransmissionis sbits;acommon valueiss =8.

Theunitsofplaintextarechainedtogether;togettheciphertextisafunctionofallprecedingplaintext.Heretheplaintextisdividedintosegmentsofs bits.

Encryption:

Theinputtotheencryptionfunctionisab-bitshiftregisterthatisinitiallysettosomeinitializationvector(IV).

Theleftmost(mostsignificant)sbitsoftheoutputoftheencryptionfunctionareXORedwiththefirstsegmentofplaintextP1toproduce thefirstunitofciphertextC1.

Thecontentsoftheshiftregisterareshiftedleftbysbits,andC1isplacedintherightmost(leastsignificant)sbitsoftheshiftregister.

Thisprocesscontinuesuntilallplaintextunits havebeenencrypted.

Decryption:

Thesameschemeisused,exceptthatthereceivedciphertextunitisXORedwiththeoutputofthe encryption functionto producetheplaintextunit.

LetMSBs(X)bedefined asthe mostsignificants bitsofX.Then

$$C_1 = P_1 \oplus \mathrm{MSB}_s[\mathrm{E}(K, \mathrm{IV})]$$

Therefore,byrearrangingterms:

$$P_1 = C_1 \oplus \mathrm{MSB}_s[\mathrm{E}(K, \mathrm{IV})]$$

Thesame reasoningholds forsubsequentstepsintheprocess.

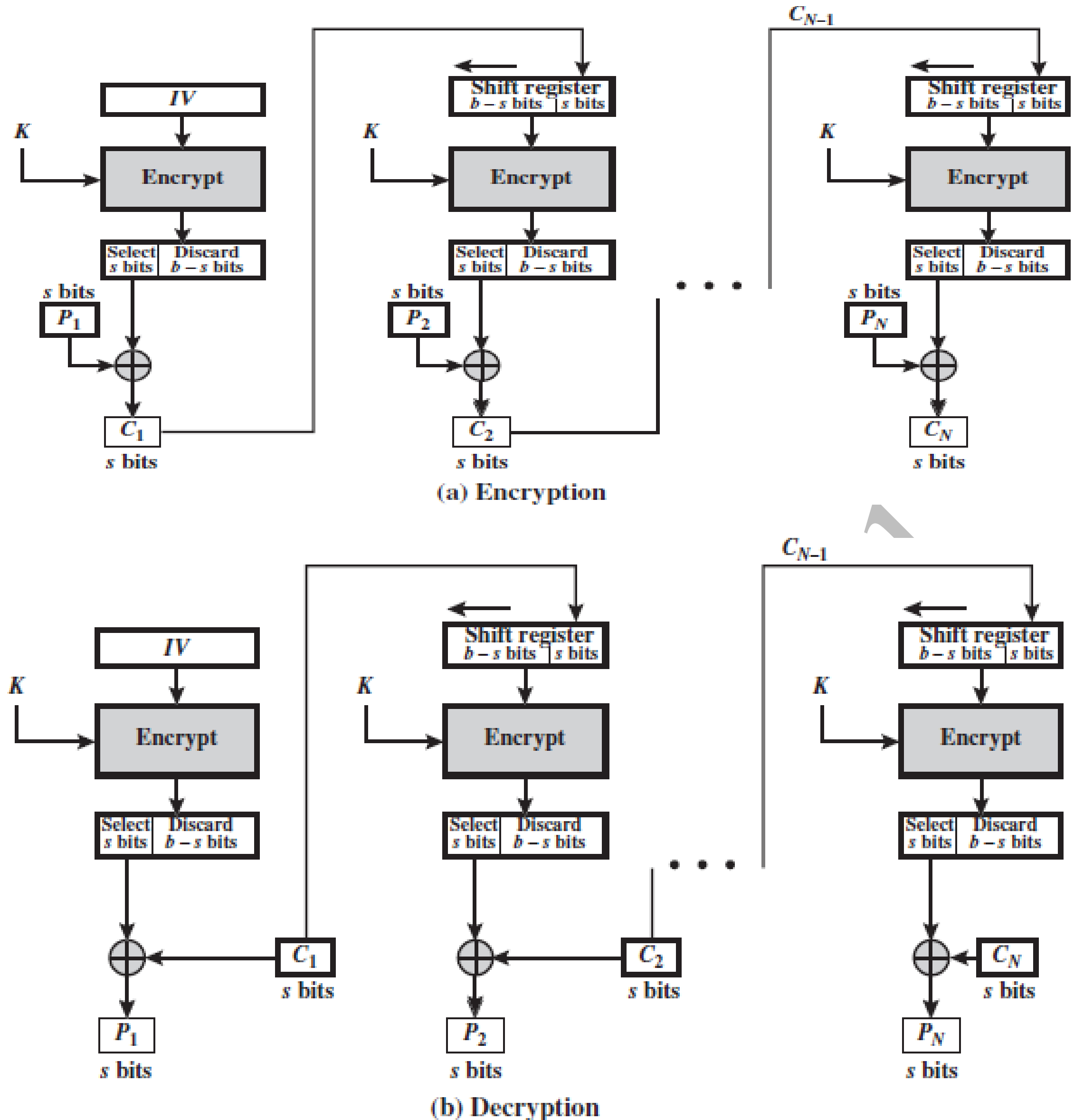


Figure2.17S-bit CipherFeedback(CFB) mode

WecandefineCFBmodeas follows

CFB	$I_1 = IV$ $I_j = \mathrm{LSB}_{b-s}(I_{j-1}) \| C_{j-1} \quad j = 2, \ldots, N$ $O_j = \mathrm{E}(K, I_j) \quad j = 1, \ldots, N$ $C_j = P_j \oplus \mathrm{MSB}_s(O_j) \quad j = 1, \ldots, N$	$I_1 = IV$ $I_j = \mathrm{LSB}_{b-s}(I_{j-1}) \| C_{j-1} \quad j = 2, \ldots, N$ $O_j = \mathrm{E}(K, I_j) \quad j = 1, \ldots, N$ $P_j = C_j \oplus \mathrm{MSB}_s(O_j) \quad j = 1, \ldots, N$

2.11.5 OutputFeedbackMode

Theoutput feedback(OFB)mode issimilarin structuretothatofCFB.

Theoutputoftheencryptionfunctionis fedbacktobecome theinputforencryptingthenextblockofplaintextasshownin figure2.18.

ComparisonbetweenOFBandCFB

In CFB,theoutputoftheXORunitis fedbacktobecomeinputfor encryptingthenext block.

Theotherdifferenceisthatthe OFBmode operatesonfullblocksofplaintextandcipher text,whereasCFBoperatesonans-bitsubset.OFBencryptioncan beexpressedasWhere

$$O_{j-1} = \mathrm{E}(K, O_{j-2})$$

wecanrewritetheencryptionexpressionas:

$$C_j = P_j \oplus \mathrm{E}(K, [C_{j-1} \oplus P_{j-1}])$$

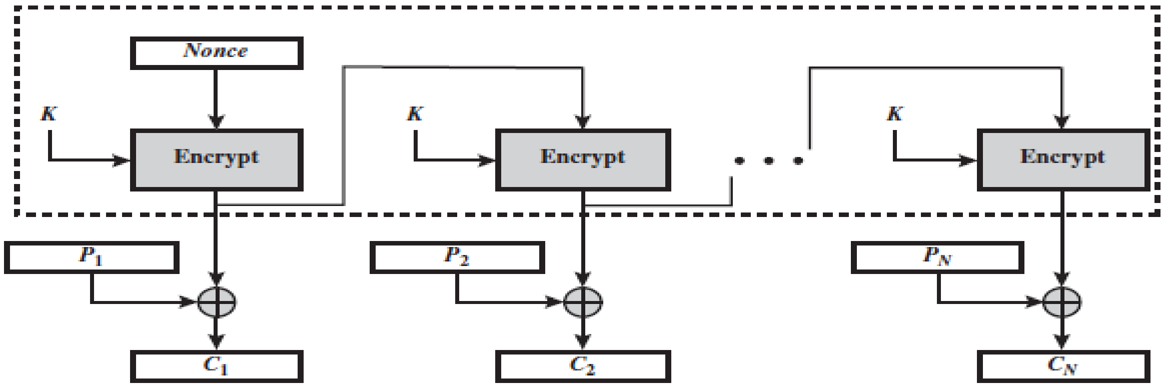
Byrearrangingterms,we can demonstratethatdecryption works.

$$P_j = C_j \oplus \mathrm{E}(K, [C_{j-1} \oplus P_{j-1}])$$

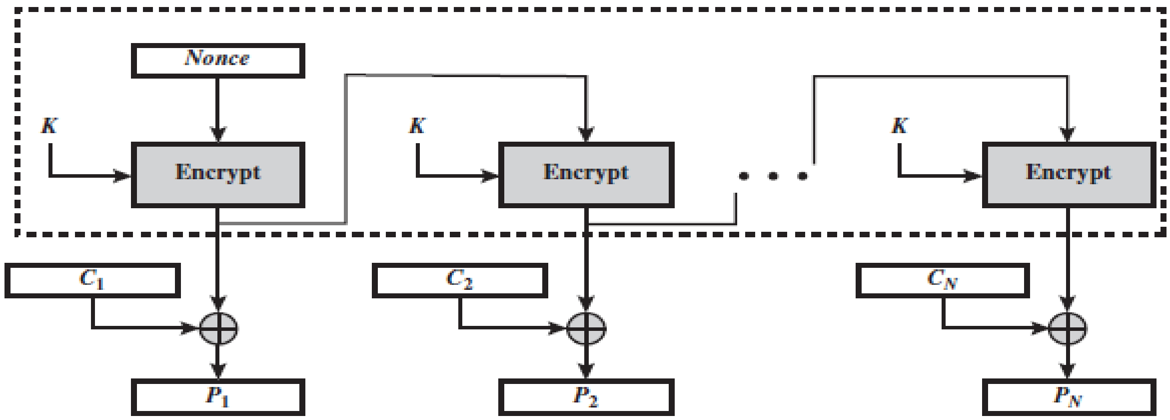
Wecan defineOFB modeasfollows.

OFB	$I_1 = Nonce$ $I_j = O_{j-1} \quad j = 2, \ldots, N$ $O_j = \mathrm{E}(K, I_j) \quad j = 1, \ldots, N$ $C_j = P_j \oplus O_j \quad j = 1, \ldots, N-1$ $C_N^* = P_N^* \oplus \mathrm{MSB}_u(O_N)$	$I_1 = Nonce$ $I_j = O_{j-1} \quad j = 2, \ldots, N$ $O_j = \mathrm{E}(K, I_j) \quad j = 1, \ldots, N$ $P_j = C_j \oplus O_j \quad j = 1, \ldots, N-1$ $P_N^* = C_N^* \oplus \mathrm{MSB}_u(O_N)$

Letthesize ofablockbeb. If the lastblockofplaintextcontains ubits(indicatedby*), withu<b,themostsignificantubitsofthelastoutputblock $O_N$areusedfortheXORoperationTheremainingb -ubitsofthelastoutputblockarediscarded.



(a) Encryption



(b) Decryption

Figure2.18OutputFeedbackMode

Advantage:

Bit errors in transmission do not propagate (i.e.) when bit errors occurs in Ci, Pi is aloneaffected

Disadvantage:

Vulnerabletomessagestreammodificationattack

2.11.6 CounterMode

Thecounter(CTR)modehasincreasedrecentlywithapplicationstoATM(asynchronous transfermode)networksecurityandIP sec(IPsecurity).

Acounterequaltotheplaintextblocksizeisused.Thecountervaluemustbedifferentfor eachplaintextblockasshowninfigure2.19.

The counter is initialized to some value and then incremented by 1 for each subsequentblock (modulo 2b, where b is the block size). For encryption, the counter is encrypted and thenXORedwiththe plaintextblocktoproduce the ciphertextblock.

For decryption, the same sequence of counter values is used, with eachencryptedcounterXORed with acipher textblocktorecover thecorrespondingplaintextblock.

Advantage:

Hardwareefficiency

- CTR can be done in parallelSoftwareefficiency
- CTRsupportsparallelfeaturepipeliningPreprocessing

Simplicity

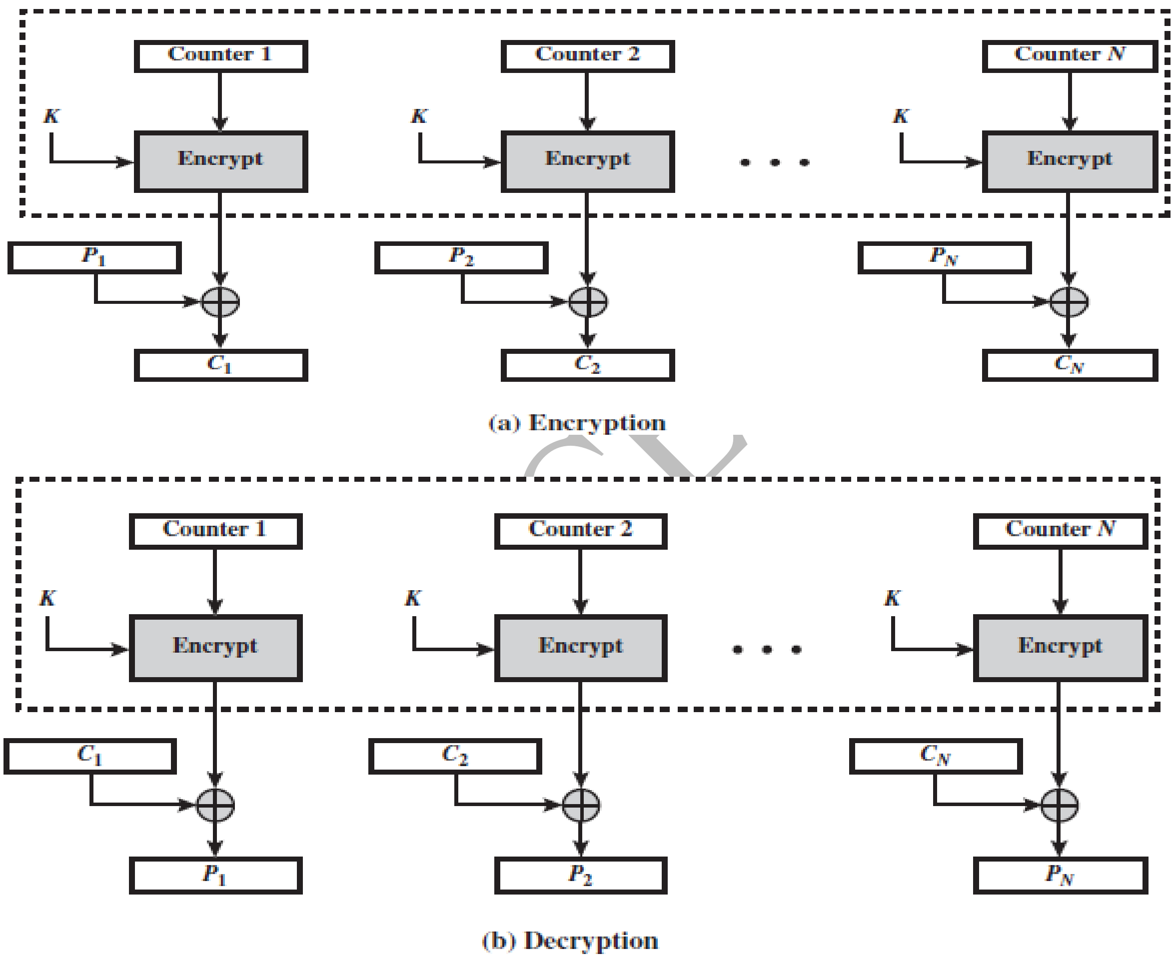


Figure2.19CounterMode

2.12 ADVANCEDENCRYPTIONSTANDARD(AES)

AES is a symmetric block cipher that is intended to replace DES as the approvedstandard for a wide range of applications. Compared to public-key ciphers such as RSA, thestructure ofAES and mostsymmetricciphers is quite complex and cannotbe explained aseasily asmanyothercryptographic,algorithms.

2.12.1 FiniteFieldArithmetic

InAES,alloperationsareperformedon8-bitbytes.Thearithmeticoperationsofaddition,multiplication,anddivisionareperformedoverthefinitefieldGF.Afieldisasetinwhichwecandoaddition,subtraction,multiplication,anddivisionwithoutleavingtheset.Divisionisdefinedwiththe followingrule:*a*/*b*=*a*(*b*-1).

An example of a finite field (one with a finite number of elements) is the set Z*p* consisting ofall the integers {0, 1, c, *p* - 1}, where *p* is a prime number and in which arithmetic is carried outmodulo*p*.

The way of defining a finite field containing 2^nelements; such a field is referred to as GF(2^n).Consider the set,*S*, of all polynomials ofdegree*n* - 1 or lesswith binary coefficients. Thus,eachpolynomialhastheform

$$f(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_1x + a_0 = \sum_{i=0}^{n-1} a_ix^i$$

Whereeach$a_i$takeson thevalue0or 1.Thereareatotalof2^ndifferentpolynomialsin*S*.For*n*=3,the 2^3=8polynomialsinthe setare

$$\begin{array}{llll} 0 & x & x^2 & x^2 + x \\ 1 & x + 1 & x^2 + 1 & x^2 + x + 1 \end{array}$$

Appropriatedefinitionof arithmetic operations,each suchset*S*isafinitefield.

Thedefinitionconsistsofthefollowingelements.

1. Arithmetic follows the ordinary rules of polynomial arithmetic using the basic rulesofalgebra withthe followingtworefinements.
2. Arithmetic on the coefficients is performed modulo 2. This is the same as theXORoperation.
3. If multiplication results in a polynomial of degree greater than *n* - 1, then the n polynomialis reduced modulo some irreducible polynomial *m*(*x*) of degree *n*. That is, we divide by*m*(*x*) and keep the remainder. For a polynomial *f*(*x*), the remainder is expressed as *r*(*x*) =*f*(*x*) mod *m*(*x*). A polynomial *m*(*x*) is called **irreducible** if and only if *m*(*x*) cannot beexpressed asaproductoftwopolynomials,bothofdegreelowerthanthatof*m*(*x*).

A polynomial in GF(2*n*) can be uniquely represented by its *n* binary coefficients(*an*-1*an*-2 c*a*0).Therefore,everypolynomialinGF(2*n*)can be represented byan*n*-bitnumber.

2.12.2 AESStructure

GeneralStructure

- Figure 2.20 shows the overall structure of the AES encryption process. The cipher takes aplaintext block size of 128 bits, or 16 bytes. The key length can be 16, 24, or32 bytes (128,192, or 256 bits). The algorithm is referred to as AES-128, AES-192, orAES-256, dependingonthekeylength.

- The input to the encryption and decryption algorithms is a single 128-bit block. The block isdepicted as a 4 * 4 square matrix of bytes. This block is copied into the **State** array, which ismodified at each stage ofencryption ordecryption. After the final stage, **State** is copied toanoutputmatrix.TheseoperationsaredepictedinFigure2.21a.Similarly,thekeyisdepictedasas quare matrixofbytes. This key isthenexpandedintoanarray of keyschedulewords.
- Below Figure 2.20 shows the expansion for the 128-bit key. Each word is four bytes, and thetotal key schedule is 44 words for the 128-bit key. Note that the ordering of bytes within amatrix is by column. The first four bytes of a 128-bit plaintext input to the encryption cipheroccupy the first column of the **in** matrix. The second four bytes occupy the second column,and so on. Similarly, the first four bytes of the expanded key, which form a word, occupy thefirst column of the **w** matrix. The cipher consists of *N* rounds, where the number of roundsdepends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14rounds fora32-bytekey(Table2.3).
- The first *N* - 1 round consist of four distinct transformation functions: Sub Bytes, Shift Rows,MixColumns,andAddRoundKey,whicharedescribedsubsequently.Thefinalroundcontain sonlythreetransformations,andthereisaninitialsingletransformation(AddRoundKey)beforethefi rstround,whichcanbeconsideredRound0.Eachtransformation takes one or more 4 * 4 matrices as input and produces a 4 * 4 matrix asoutput Figure 5.1 shows that the output of each round is a 4 * 4 matrix, with the output of thefinalroundbeingthe ciphertext.

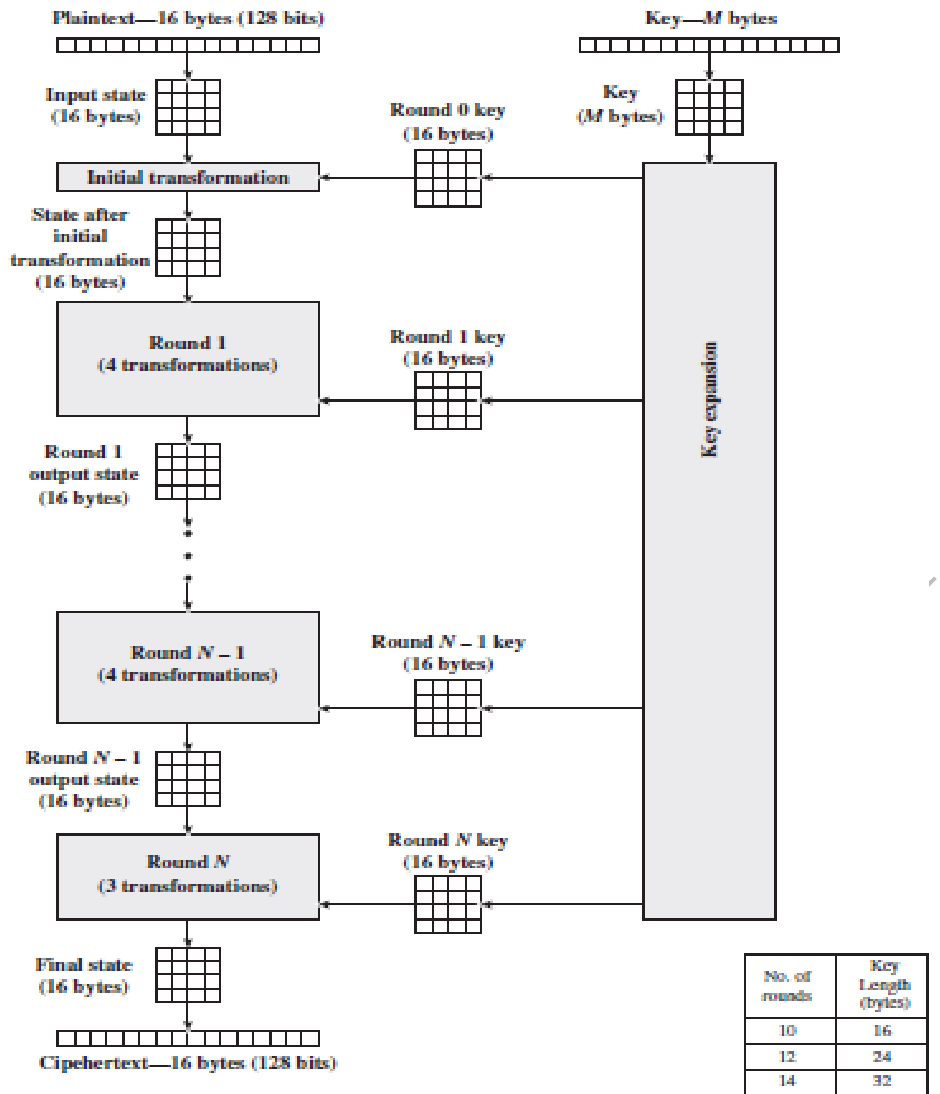


Figure2.20 AES EncryptionProcess

Key Size (words/bytes/bits)	4/16/128	6/24/192	8/32/256
Plaintext Block Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Number of Rounds	10	12	14
Round Key Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Expanded Key Size (words/bytes)	44/176	52/208	60/240

Table2.3AESParameters

2.12.3 DetailedStructure

BelowFigure 2.20showstheAEScipher showsthesequenceoftransformationsineach roundandshowingthe correspondingdecryptionfunction.

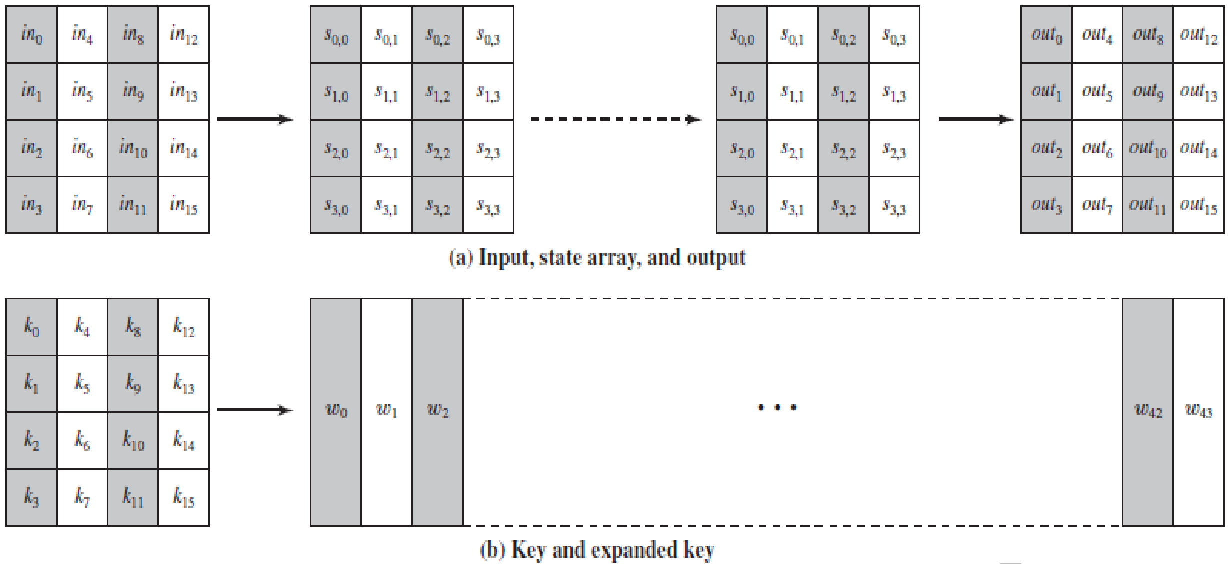


Fig:2.21

DetailAESstructureOveralldetailaboutAES structure.

1. It is not a Feistel structure. Recall that, in the classic Feistel structure, half of the datablock is used to modify the other half of the data block and then the halves are swapped.AES instead processes the entire data block as a single matrix during each round usingsubstitutionsandpermutation.
2. The key that is provided as input is expanded into an array of forty-four 32-bitwords, **w**[*i*].Four distinct words (128 bits) serve as a round key for each round as shown in figure2.22;
3. Fourdifferentstagesareused, one ofpermutationandthreeof substitution:
   - **Substitute bytes:** Uses an S-box to perform a byte-by-byte substitution oftheblock
   - **ShiftRows:**Asimple permutation
   - **MixColumns:**AsubstitutionthatmakesuseofarithmeticoverGF(28)
   - **AddRoundKey:** A simple bitwise XOR of the current block with a portion oftheexpanded key
4. The structure is quite simple.Forboth encryption and decryptionas shown in figure2.22, the cipher begins with an AddRoundKey stage, followed by nine rounds that eachincludesall fourstages,followedby atenth round ofthreestages.
5. Only the AddRoundKey stage makes use of the key. The AddRoundKey stage wouldprovidenosecuritybecausetheydonotusethekey.Wecanviewthecipherasalternatingo perationsofXORencryption(AddRoundKey)ofablock,followedbyscrambling of the block (the other three stages), followed by XOR encryption, and so on.This scheme isbothefficientandhighlysecure.

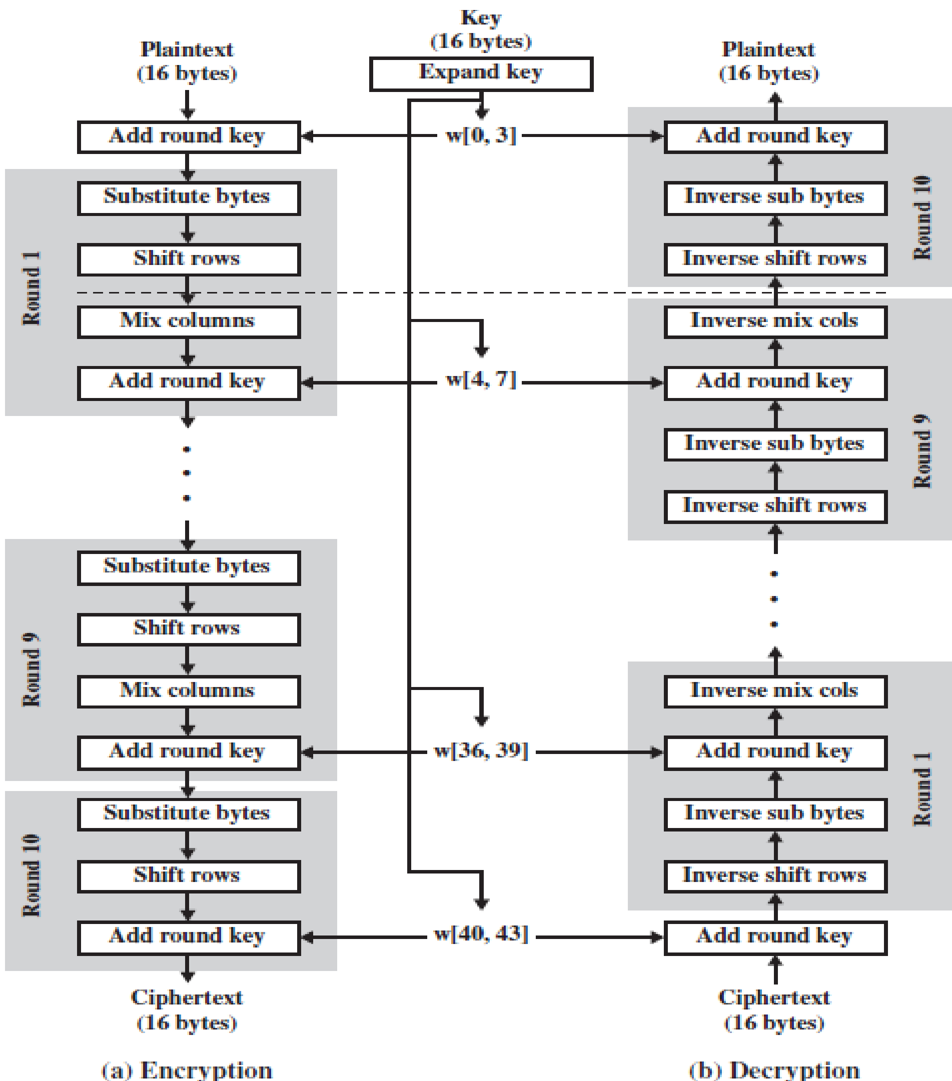


Fig2.22AESEncryptionandDecryption

6. Each stage is easily reversible. For the Substitute Byte, ShiftRows, and MixColumnsstages, an inverse function is used in the decryption algorithm. For the AddRoundKeystage,theinverseisachievedbyXORingthesame roundkeytotheblock,usingthe result that.

$$A \oplus B \oplus B = A$$

7. The decryption algorithm makes use of the expanded key in reverse order. However, thedecryptionalgorithmisnotidenticaltotheencryptionalgorithm.Thisisaconsequenceofthep articularstructureofAES.

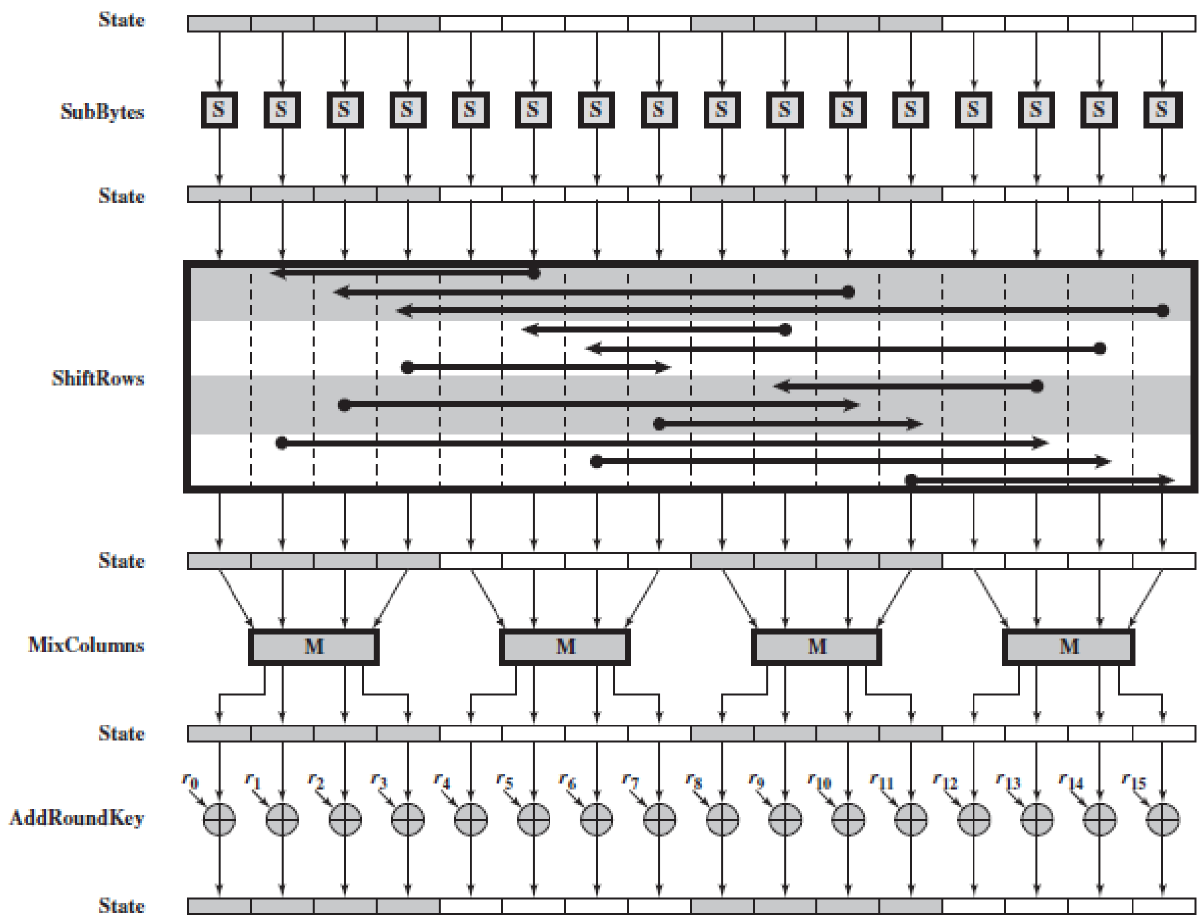


Fig2.23AESEncryptionRound

8. Once itisestablishedthatallfour stagesarereversible,it iseasytoverifythatdecryptiondoesrecovertheplaintext.

9. Thefinalroundofbothencryptionanddecryption consists ofonly threestages.Again,thisisaconsequenceof theparticularstructure ofAES andisrequired,tomake thecipherreversible

2.12.4 AESTransformationFunctions

Four transformations used in AES. For each stage, we describe the forward (encryption)algorithm,theinverse (decryption)algorithm,andtherationaleforthestage.

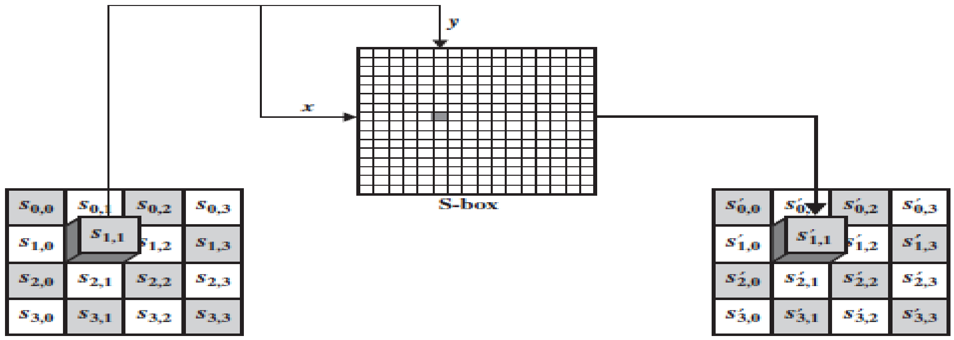
SubstituteBytesTransformation

Type1: ForwardandInverseTransformations*:*

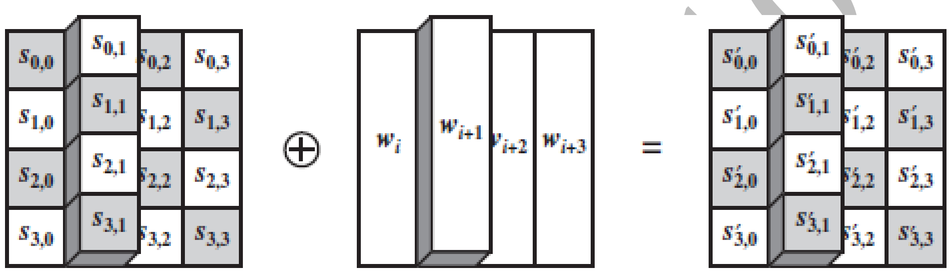
***The*forwardsubstitutebytetransformation**,calledSubBytes,isasimpletablelookup (Figure2.24a). AESdefines a16* 16 matrix ofbytevalues,calledanS-box thatcontainsa permutationofallpossible256 8-bitvalues.

Each individual byte of**State** is mapped into a new byte in the following way: Theleftmost 4 bits of the byte are used as a row value and the rightmost 4 bits are used as a columnvalue. These row and column values serve as indexes into the S-box to select a unique8-bitoutputvalueasshownin figure 2.25.

For example, the hexadecimal value {95} references row 9, column 5 of the S-box, whichcontains thevalue{2A}.Accordingly,thevalue{95}ismappedinto thevalue{2A}.



(a) Substitute byte transformation



(b) Add round key transformation

Figure2.24AESBytelevelOperations

x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

(a) S-box

x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

(b) Inverse S-box

Figure2.25 AESS-Boxes

Here isanexample ofthe SubBytestransformation:

EA	04	65	85
83	45	5D	96
5C	33	98	B0
F0	2D	AD	C5

→

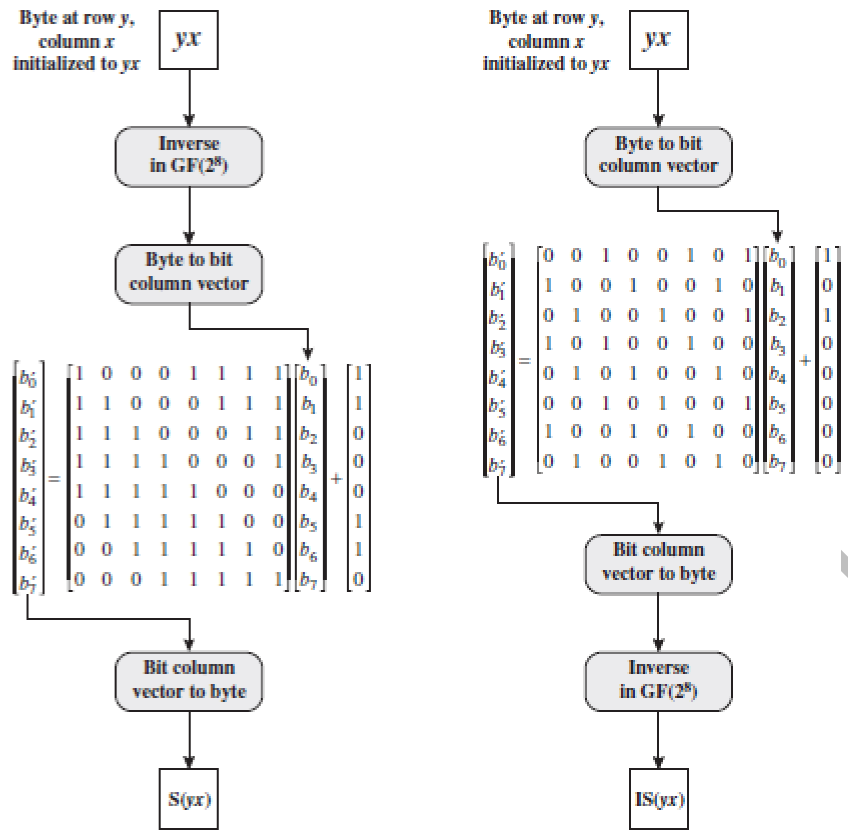
87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

TheS-box isconstructedinthefollowingfashion(Figure2.26a).

1. Initialize the S-box with the byte values in ascending sequence row by row. The first rowcontains {00}, {01}, {02}, c, {0F}; the second row contains {10}, {11}, etc.; and so on. Thus, thevalueofthe byteatrow*y*,column*x*is{*yx*}.
2. MapeachbyteintheS-boxtoitsmultiplicativeinverseinthefinitefieldGF(28);thevalue {00}is mappedtoitself.
3. Consider that each byte in the S-box consists of 8 bits labeled (*b*7, *b*6, *b*5, *b*4, *b*3,*b*2, *b*1, *b*0).Apply the followingtransformationtoeach bitofeachbyteinthe S-box:

$$b'_i = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i$$

Where*ci*is the*i*thbit ofbyte*c*withthevalue{63}; that is,($c_7c_6c_5c_4c_3c_2c_1c_0$)=(01100011).
Theprime(„)indicates thatthevariableis tobe updatedbythevalueontheright.



(a) Calculation of byte at row y, column x of S-box

(a) Calculation of byte at row y, column x of IS-box

Figure2.26ConstructionofS-Box andIS-Box

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

TheAESstandard depictsthistransformationinmatrixformasfollows.

- Inordinarymatrixmultiplication,eachelementintheproductmatrixisthesumofproducts of the elements ofone row and one column.Each element in the productmatrix isthebitwiseXORofproductsofelementsofone rowandonecolumn.
- Asanexample,considertheinputvalue{95}.ThemultiplicativeinverseinGF(28)is $\{95\}^{-1}$={8A},whichis10001010inbinary.UsingaboveEquation

$$\begin{bmatrix} 1&0&0&0&1&1&1&1\\ 1&1&0&0&0&1&1&1\\ 1&1&1&0&0&0&1&1\\ 1&1&1&1&0&0&0&1\\ 1&1&1&1&1&0&0&0\\ 0&1&1&1&1&1&0&0\\ 0&0&1&1&1&1&1&0\\ 0&0&0&1&1&1&1&1 \end{bmatrix} \begin{bmatrix} 0\\1\\0\\1\\0\\0\\0\\1 \end{bmatrix} \oplus \begin{bmatrix} 1\\1\\0\\0\\0\\1\\1\\0 \end{bmatrix} = \begin{bmatrix} 1\\0\\0\\1\\0\\0\\1\\0 \end{bmatrix} \oplus \begin{bmatrix} 1\\1\\0\\0\\0\\1\\1\\0 \end{bmatrix} = \begin{bmatrix} 0\\1\\0\\1\\0\\1\\0\\0 \end{bmatrix}$$

Theresultis {2A},which shouldappearinrow{09}column {05}of theS-box.

Type2:InverseSubstituteByteTransformation:

The **inverse substitute byte transformation**, called InvSubBytes, For example, that theinput {2A}produces the output {95}, and the input {95} to the S-box produces {2A}. The inverseS-boxisconstructedbyapplyingtheinverseofthetransformationisfollowedbytakingthe

$$b'_i = b_{(i+2) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus d_i$$

multiplicativeinverse inGF(28).The inversetransformationis

where byte$d$={05},or00000101.Wecandepictthistransformationas follows.

$$\begin{bmatrix} b'_0\\ b'_1\\ b'_2\\ b'_3\\ b'_4\\ b'_5\\ b'_6\\ b'_7 \end{bmatrix} = \begin{bmatrix} 0&0&1&0&0&1&0&1\\ 1&0&0&1&0&0&1&0\\ 0&1&0&0&1&0&0&1\\ 1&0&1&0&0&1&0&0\\ 0&1&0&1&0&0&1&0\\ 0&0&1&0&1&0&0&1\\ 1&0&0&1&0&1&0&0\\ 0&1&0&0&1&0&1&0 \end{bmatrix} \begin{bmatrix} b_0\\ b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7 \end{bmatrix} + \begin{bmatrix} 1\\0\\1\\0\\0\\0\\0\\0 \end{bmatrix}$$

InvSubBytesistheinverseofSubBytes,labelthematricesinsubBytesandInvSubBytes as **X** and **Y**, respectively, and the vector versions of constants c and d as **C** and **D**,respectively.For some8-bitvector**B,**becomes $\mathbf{B}' = \mathbf{XB} \oplus \mathbf{C}$.Weneedtoshowthat $\mathbf{Y}(\mathbf{XB} \oplus \mathbf{C}) \oplus \mathbf{D} = \mathbf{B}$**.**Tomultiplyout,wemustshow $\mathbf{YXB} \oplus \mathbf{YC} \oplus \mathbf{D} = \mathbf{B}$**.**Thisbecomes

$$\begin{bmatrix} 0&0&1&0&0&1&0&1\\ 1&0&0&1&0&0&1&0\\ 0&1&0&0&1&0&0&1\\ 1&0&1&0&0&1&0&0\\ 0&1&0&1&0&0&1&0\\ 0&0&1&0&1&0&0&1\\ 1&0&0&1&0&1&0&0\\ 0&1&0&0&1&0&1&0 \end{bmatrix} \begin{bmatrix} 1&0&0&0&1&1&1&1\\ 1&1&0&0&0&1&1&1\\ 1&1&1&0&0&0&1&1\\ 1&1&1&1&0&0&0&1\\ 1&1&1&1&1&0&0&0\\ 0&1&1&1&1&1&0&0\\ 0&0&1&1&1&1&1&0\\ 0&0&0&1&1&1&1&1 \end{bmatrix} \begin{bmatrix} b_0\\ b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7 \end{bmatrix} \oplus$$

$$\begin{bmatrix} 0&0&1&0&0&1&0&1\\ 1&0&0&1&0&0&1&0\\ 0&1&0&0&1&0&0&1\\ 1&0&1&0&0&1&0&0\\ 0&1&0&1&0&0&1&0\\ 0&0&1&0&1&0&0&1\\ 1&0&0&1&0&1&0&0\\ 0&1&0&0&1&0&1&0 \end{bmatrix} \begin{bmatrix} 1\\1\\0\\0\\0\\1\\1\\0 \end{bmatrix} \oplus \begin{bmatrix} 1\\0\\1\\0\\0\\0\\0\\0 \end{bmatrix} =$$

$$\begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1 \end{bmatrix} \begin{bmatrix} b_0\\ b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7 \end{bmatrix} \oplus \begin{bmatrix} 1\\0\\1\\0\\0\\0\\0\\0 \end{bmatrix} \oplus \begin{bmatrix} 1\\0\\1\\0\\0\\0\\0\\0 \end{bmatrix} = \begin{bmatrix} b_0\\ b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7 \end{bmatrix}$$

Wehavedemonstrated that**YX**equals theidentitymatrix,andthe**YC=D**,so that**YC**$\oplus$**D** equalsthenull vector.

Type3:ShiftRows Transformation

ForwardandInverseShiftRowsTransformations*:*

The forward shift row transformation, called Shift Rows, is depicted in Figure 2.27.The first row of **State** is not altered. For the second row, a 1-byte circular left shift is performed.For the third row, a 2-bytecircular left shift is performed. For the fourth row, a 3-byte circular leftshiftisperformed. ThefollowingisanexampleofShiftRows

87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

→

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

Figure2.27ForwardShiftRowTransformation

The inverse shift row transformation, called InvShiftRows, performs the circular shiftsin the opposite direction for each of the last three rows, with a 1-byte circular right shift for thesecondrow,and as showninfigure2.28

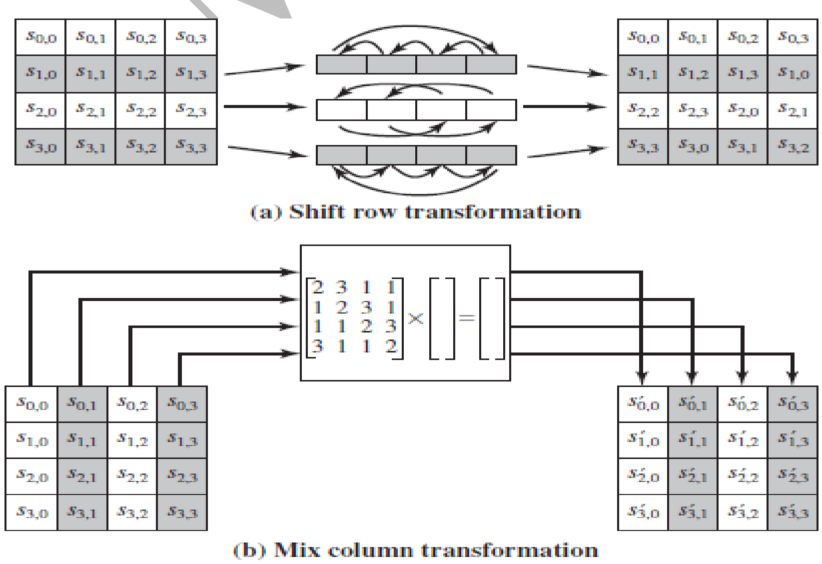


Figure2.28AESRowandColumnOperations

Type4: MixColumnsTransformation

Forward and Inverse Transformations*: The* forward mix column transformation,calledMixColumns,operatesoneachcolumnindividually.Eachbyteofacolumnisma ppedinto a new value that is a function of all four bytes in that column. The transformation can bedefinedbythefollowingmatrixmultiplicationon**State**

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

Each elementin the productmatrix is the sumofproducts ofelements ofone rowandonecolumn.Inthiscase,theindividualadditionsandmultiplicationsareperformedinGF(2^8).

$$\begin{aligned} s'_{0,j} &= (2 \cdot s_{0,j}) \oplus (3 \cdot s_{1,j}) \oplus s_{2,j} \oplus s_{3,j} \\ s'_{1,j} &= s_{0,j} \oplus (2 \cdot s_{1,j}) \oplus (3 \cdot s_{2,j} \oplus s_{3,j} \\ s'_{2,j} &= s_{0,j} \oplus s_{1,j} \oplus (2 \cdot s_{2,j}) \oplus (3 \cdot s_{3,j}) \\ s'_{3,j} &= (3 \cdot s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus (2 \cdot s_{3,j}) \end{aligned}$$

The MixColumns transformation on a single column of **State** can be expressed asThefollowingisanexampleofMixColumns:

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

→

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

TheMixColumnstransformationonthefirstcolumn, weneed toshowthat

$$\begin{array}{lllll} (\{02\} \cdot \{87\}) & \oplus (\{03\} \cdot \{6E\}) & \oplus \{46\} & \oplus \{A6\} & = \{47\} \\ \{87\} & \oplus (\{02\} \cdot \{6E\}) & \oplus (\{03\} \cdot \{46\}) & \oplus \{A6\} & = \{37\} \\ \{87\} & \oplus \{6E\} & \oplus (\{02\} \cdot \{46\}) & \oplus (\{03\} \cdot \{A6\}) & = \{94\} \\ (\{03\} \cdot \{87\}) & \oplus \{6E\} & \oplus \{46\} & \oplus (\{02\} \cdot \{A6\}) & = \{ED\} \end{array}$$

For thefirstequation,we have{02}.{87} =(00001110)$\oplus$(00011011)=(0001 0101) and

{03}.{6E}={6E}$\oplus$({02}.{6E})=(01101110)$\oplus$(11011100)=(10110010)then

$$\begin{array}{lcl} \{02\} \cdot \{87\} & = & 0001\ 0101 \\ \{03\} \cdot \{6E\} & = & 1011\ 0010 \\ \{46\} & = & 0100\ 0110 \\ \{A6\} & = & \underline{1010\ 0110} \\ & & 0100\ 0111 = \{47\} \end{array}$$

The **inverse mix column transformation**, called InvMixColumns, is defined bythefollowingmatrixmultiplication:

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

The**inverse**ofEquationneedtoshow

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix}$$

That is, the inverse transformation matrix times the forward transformation matrixequals theidentity matrix.To verifythefirstcolumnofaboveEquation.
For thefirstequation,wehave {0E}.{02} =00011100and{09}.{03} ={09}$\oplus${09}.{02}= 00001001$\oplus$00010010=00011011then

$$\begin{aligned} \{0E\} \cdot \{02\} &= 00011100 \\ \{0B\} &= 00001011 \\ \{0D\} &= 00001101 \\ \{09\} \cdot \{03\} &= \underline{00011011} \\ & 00000001 \end{aligned}$$

Theencryptionwasdeemedmoreimportantthan decryption fortworeasons:

1. FortheCFBandOFBciphermodesonlyencryptionisused.
2. AES canbeusedtoconstructamessageauthenticationcodeand for this,only encryptionisused.

Type 5: AddRoundKey

TransformationForwardandInverseTra

nsformations

In the **forwardaddroundkeytransformation**,calledAddRoundKey,the128bitsof**State**arebitwiseXORed withthe128bitsoftheround key.
Theoperationisviewedasacolumnwiseoperationbetweenthe4bytesofa**State**column andonewordoftheroundkey;itcan alsobeviewedasabyte-leveloperation.
Thefollowingis an example ofAddRoundKey:

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

$\oplus$

AC	19	28	57
77	FA	D1	5C
66	DC	29	00
F3	21	41	6A

=

EB	59	8B	1B
40	2E	A1	C3
F2	38	13	42
1E	84	E7	D6

Thefirstmatrix is**State**,andthe secondmatrix istheroundkey.

The **inverse add round key transformation** is identical to the forward addround keytransformation,becausetheXORoperationisitsowninverse.

TheFigure2.29isanotherviewofasingle roundofAES, emphasizingthemechanismsandinputsofeach transformation.

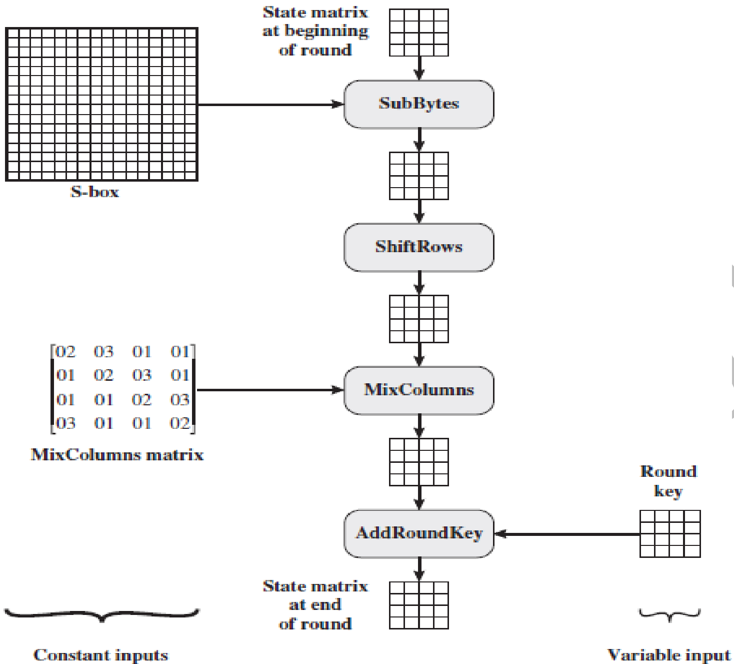


Fig 2.29 AES Key

ExpansionType6:KeyExpansionAlgorithm

The AES key expansion algorithm takes as input a four-word (16-byte) key and producesa linear array of 44 words (176 bytes). This is sufficient to provide a four word round key for theinitialAddRoundKeystage andeach ofthe10roundsofthecipher.

Each added word **w**[i]depends on the immediately preceding word, **w**[i - 1], and the wordfour positions back, **w**[i - 4]. In three out of four cases, a simple XOR is used. For a word whosepositioninthe**w**arrayisamultipleof4,amore complexfunctionisused.

Figure2.30illustratesthegenerationoftheexpandedkey,usingthesymbolgtorepresentthatcomplexfunction.Thefunctiongconsistsofthefollowingsubfunctions

```
KeyExpansion (byte key[16], word w[44])
{
    word temp
    for (i = 0; i < 4; i++)    w[i] = (key[4*i], key[4*i+1],
                                      key[4*i+2],
                                      key[4*i+3]);
     for (i = 4; i < 44; i++)
     {
      temp = w[i - 1];
      if (i mod 4 = 0)    temp = SubWord (RotWord (temp))
                                 ⊕ Rcon[i/4];
      w[i] = w[i-4]  ⊕  temp
     }
}
```

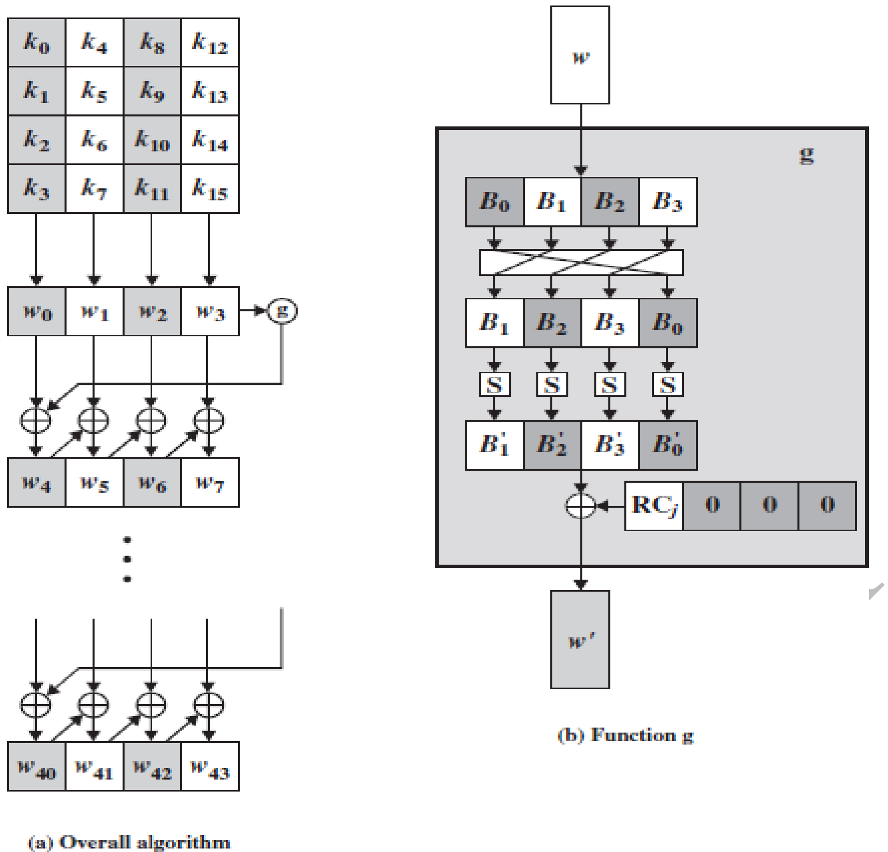


Figure2.30KeyExpansionAlgorithm

1. RotWordperforms aone-bytecircular leftshiftonaword.This meansthatainputword [B0,B1,B2,B3]istransformedinto[B1,B2,B3,B0].

2. SubWord performsabyte substitution on eachbyteofitsinputword,usingtheS-box.

3. Theresultofsteps1and2is XORedwitha roundconstant,Rcon[j].

Theroundconstantisa wordinwhichthethree rightmostbytesarealways0.Thus,theeffectofanXOR ofawordwithRconistoonly performanXORon theleftmostbyte oftheword.Theroundconstantisdifferentforeachroundand isdefined asRcon[j] = (RC[j],0,0,0),

with RC[1]= 1,RC[j] =2#RC[j-1]and with multiplication definedoverthefieldGF(28).ThevaluesofRC[j]in hexadecimalare

j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

Forexample,suppose thatthe round keyforround8is

EAD27321B58DBAD2312BF5607F 8D292F

Thenthefirst4bytes (firstcolumn) oftheroundkeyforround9are calculatedasfollows:

i (decimal)	temp	After RotWord	After SubWord	Rcon (9)	After XOR with Rcon	w[i−4]	w[i] = temp ⊕ w[i−4]
36	7F8D292F	8D292F7F	5DA515D2	1B000000	46A515D2	EAD27321	AC7766F3

AnAESExample

For this example, the plaintext is a hexadecimal palindrome. The plaintext,key, and resultingciphertextare

Plaintext:	0123456789abcdeffedcba9876543210
Key:	0f1571c947d9e8590cb7add6af7f6798
Ciphertext:	ff0b844a0853bf7c6934ab4364148fb9

Results

Table 2.4shows theexpansionofthe16-byte key into10round keys.The processis formedwordbyword,with eachfour-byte wordoccupyingone column oftheword round-keymatrix.

Key Words	Auxiliary Function
w0 = 0f 15 71 c9 w1 = 47 d9 e8 59 w2 = 0c b7 ad d6 w3 = af 7f 67 98	RotWord(w3) = 7f 67 98 af = x1 SubWord(x1) = d2 85 46 79 = y1 Rcon(1) = 01 00 00 00 y1 ⊕ Rcon(1) = d3 85 46 79 = z1
w4 = w0 ⊕ z1 = dc 90 37 b0 w5 = w4 ⊕ w1 = 9b 49 df e9 w6 = w5 ⊕ w2 = 97 fe 72 3f w7 = w6 ⊕ w3 = 38 81 15 a7	RotWord(w7) = 81 15 a7 38 = x2 SubWord(x2) = 0c 59 5c 07 = y2 Rcon(2) = 02 00 00 00 y2 ⊕ Rcon(2) = 0e 59 5c 07 = z2
w8 = w4 ⊕ z2 = d2 c9 6b b7 w9 = w8 ⊕ w5 = 49 80 b4 5e w10 = w9 ⊕ w6 = de 7e c6 61 w11 = w10 ⊕ w7 = e6 ff d3 c6	RotWord(w11) = ff d3 c6 e6 = x3 SubWord(x3) = 16 66 b4 83 = y3 Rcon(3) = 04 00 00 00 y3 ⊕ Rcon(3) = 12 66 b4 8e = z3
w12 = w8 ⊕ z3 = c0 af df 39 w13 = w12 ⊕ w9 = 89 2f 6b 67 w14 = w13 ⊕ w10 = 57 51 ad 06 w15 = w14 ⊕ w11 = b1 ae 7e c0	RotWord(w15) = ae 7e c0 b1 = x4 SubWord(x4) = e4 f3 ba c8 = y4 Rcon(4) = 08 00 00 00 y4 ⊕ Rcon(4) = ec f3 ba c8 = 4

Table2.4 Expansion ofthe16-byte keyinto 10roundkeys

The left-hand column shows the four round-key words generated for each round. Theright-hand column shows the steps used to generate the auxiliary word used in key expansion.Thekey itselfservingasthe round keyforround0.

Next, Table 2.5 shows the progression of **State** through the AES encryption process.The first column shows the value of **State** at the start of a round. For the first row, **State** is justthe matrix arrangement of the plaintext. The second, third, and fourth columns show the value of**State**forthatroundaftertheSubBytes,ShiftRows,andMixColumnstransformations,respectively.Thefifthcolumnshowstheroundkey.

Key Words	Auxiliary Function
w16 = w12 ⊕ z4 = 2c 5c 65 f1 w17 = w16 ⊕ w13 = a5 73 0e 96 w18 = w17 ⊕ w14 = f2 22 a3 90 w19 = w18 ⊕ w15 = 43 8c dd 50	RotWord(w19) = 8c dd 50 43 = x5 SubWord(x5) = 64 c1 53 1a = y5 Rcon(5) = 10 00 00 00 y5 ⊕ Rcon(5) = 74 c1 53 1a = z5
w20 = w16 ⊕ z5 = 58 9d 36 eb w21 = w20 ⊕ w17 = fd ee 38 7d w22 = w21 ⊕ w18 = 0f cc 9b ed w23 = w22 ⊕ w19 = 4c 40 46 bd	RotWord (w23) = 40 46 bd 4c = x6 SubWord (x6) = 09 5a 7a 29 = y6 Rcon(6) = 20 00 00 00 y6 ⊕ Rcon(6) = 29 5a 7a 29 = z6
w24 = w20 ⊕ z6 = 71 c7 4c c2 w25 = w24 ⊕ w21 = 8c 29 74 bf w26 = w25 ⊕ w22 = 83 e5 ef 52 w27 = w26 ⊕ w23 = cf a5 a9 ef	RotWord (w27) = a5 a9 ef cf = x7 SubWord (x7) = 06 d3 bf 8a = y7 Rcon (7) = 40 00 00 00 y7 ⊕ Rcon(7) = 46 d3 df 8a = z7
w28 = w24 ⊕ z7 = 37 14 93 48 w29 = w28 ⊕ w25 = bb 3d e7 f7 w30 = w29 ⊕ w26 = 38 d8 08 a5 w31 = w30 ⊕ w27 = f7 7d a1 4a	RotWord (w31) = 7d a1 4a f7 = x8 SubWord (x8) = ff 32 d6 68 = y8 Rcon (8) = 80 00 00 00 y8 ⊕ Rcon(8) = 7f 32 d6 68 = z8
w32 = w28 ⊕ z8 = 48 26 45 20 w33 = w32 ⊕ w29 = f3 1b a2 d7 w34 = w33 ⊕ w30 = cb c3 aa 72 w35 = w34 ⊕ w32 = 3c be 0b 3	RotWord (w35) = be 0b 38 3c = x9 SubWord (x9) = ae 2b 07 eb = y9 Rcon (9) = 1B 00 00 00 y9 ⊕ Rcon (9) = b5 2b 07 eb = z9
w36 = w32 ⊕ z9 = fd 0d 42 cb w37 = w36 ⊕ w33 = 0e 16 e0 1c w38 = w37 ⊕ w34 = c5 d5 4a 6e w39 = w38 ⊕ w35 = f9 6b 41 56	RotWord (w39) = 6b 41 56 f9 = x10 SubWord (x10) = 7f 83 b1 99 = y10 Rcon (10) = 36 00 00 00 y10 ⊕ Rcon (10) = 49 83 b1 99 = z10
w40 = w36 ⊕ z10 = b4 8e f3 52 w41 = w40 ⊕ w37 = ba 98 13 4e w42 = w41 ⊕ w38 = 7f 4d 59 20 w43 = w42 ⊕ w39 = 86 26 18 76	

Table2.5 progressionof Statethroughthe AESencryption process

2.13 RC4ALGORITHM

RC4 is an encryption algorithm created in 1987 by Ronald Rivest of RSA Security. It is astream cipher (figure 2.31), which means that each digit or character is encrypted one at a time.Acipherisamessage thathasbeenencoded.

A key input is pseudorandom bit generator that produces a stream 8-bit number that isunpredictablewithoutknowledgeofinputkey.

The output of the generator is called key-stream, is combined one byte at a time with theplaintextstreamcipherusingX-OR operation.

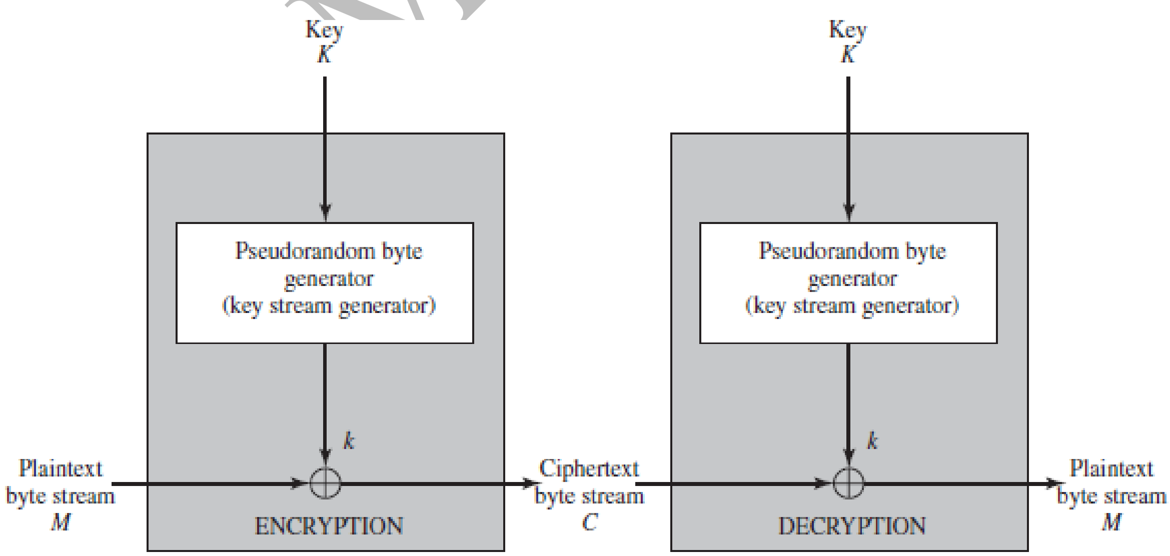


Figure2.31StreamCipherDiagram

Example

RC4 Encryption			RC4 Decryption	
10011000	Plaintext		11001000	Ciphertext
⊕ 01010000	Key Stream		⊕ 01010000	Key Stream
11001000	Ciphertext		10011000	Plaintext

2.13.1KeyGenerationAlgorithm

A variable-length key from 1 to 256 byte is used to initialize a 256-byte state vector S,with elements S[0] to S[255]. For encryption and decryption, a byte k is generated from S byselectingoneofthe255entriesinasystematicfashion,thentheentriesinSarepermutedagain (Figure2.32).

InitializationofS

The entries of S are set equal to the values from 0 to 255 in ascending orders, atemporary vector T, is created. If the length of the key k is 256 bytes, then k is assigned to T.Otherwise, for a key with length(klen) bytes, the first klen elements of T as copied from K andthenKisrepeatedasmany timesasnecessarytofillT.

```
//Initializationf
or
i=0 to
255doS[i]=i;T[i]=K[im
odklen];
```

Next, use T to produce the initial permutation of S. Starting with S[0] to S[255], and foreach S[i] algorithm swap it with another byte in S according to a scheme dictated by T[i], but Swillstillcontainvaluesfrom0to 255:

```
// Initial Permutation of
Sj=0;
for i =
0to255do
 {
 j =(j+S[i]+T[i])mod256;
 Swap(S[i],S[j]);
 }
```

Pseudorandomgenerationalgorithm(StreamGeneration)

Once the vector S is initialized, the input key will not be used. In this step, for each S[i]algorithmswapitwithanotherbyteinSaccordingtoaschemedictatedbythecurrentconfigurationofS . After reachingS[255] theprocesscontinues, startingfromS[0] again

//Stream

Generationi,j=0;

while(true)

i=(i+1)mod256;

j = (j + S[i]) mod

256;Swap(S[i],S[j]);

t = (S[i] + S[j]) mod

256;k=S[t];

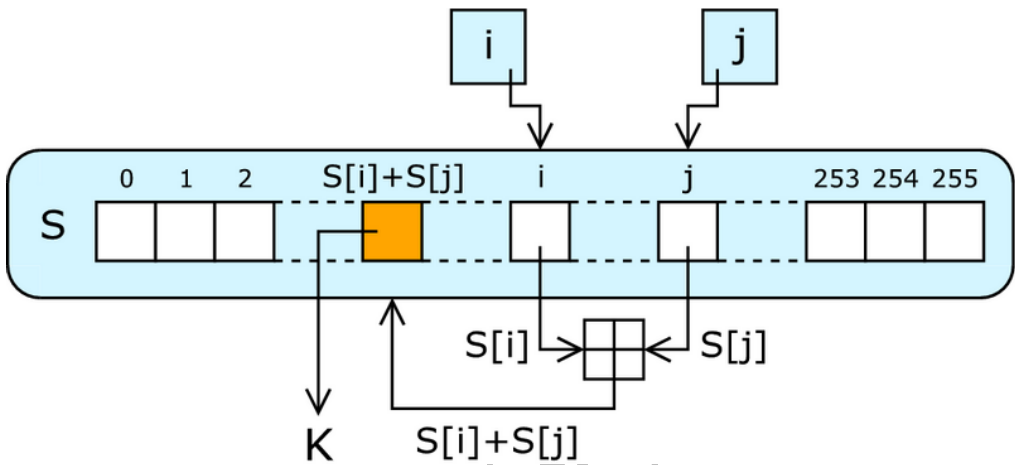


Figure2.32PRGAAlgorithm

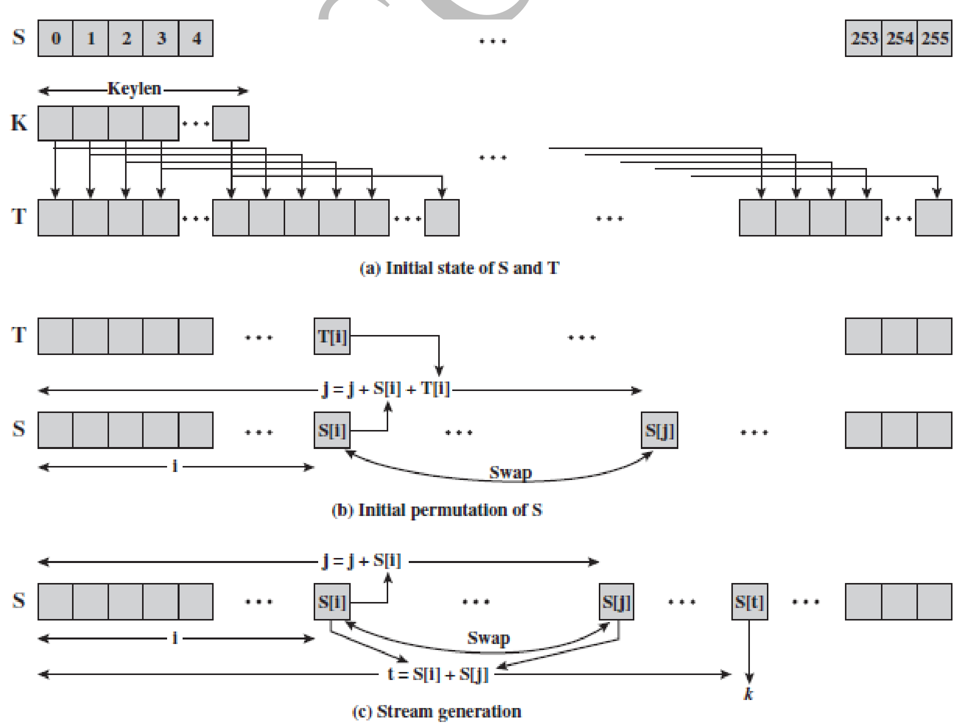


Figure2.33RC4Algorithm

EncryptusingXOR

Toencrypt,XOR thevalue kwith thenextbyteof plaintext.

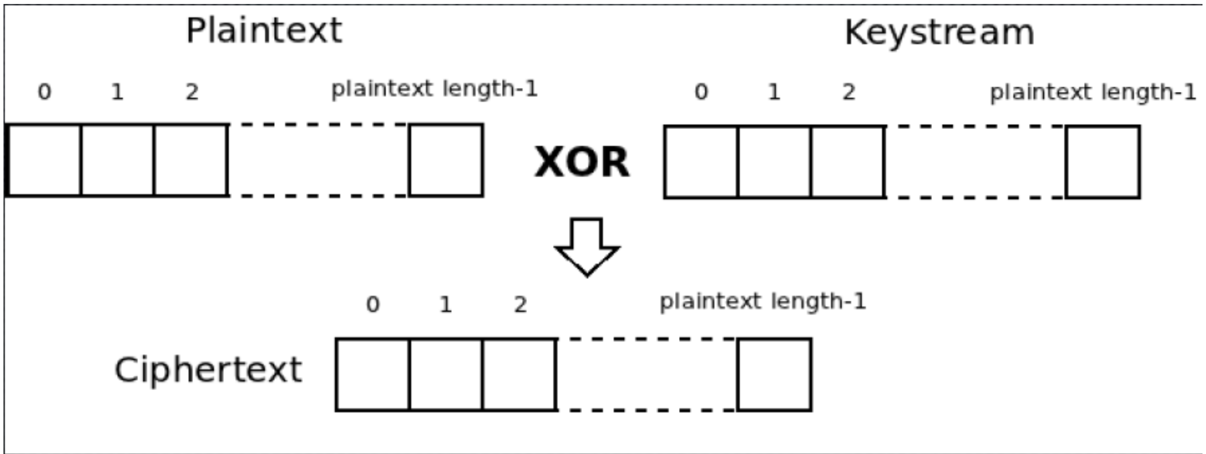


Figure 2.34RC4 Encryption

DecryptusingXOR

Todecrypt,XORthevaluekwiththenextbyteofciphertext.

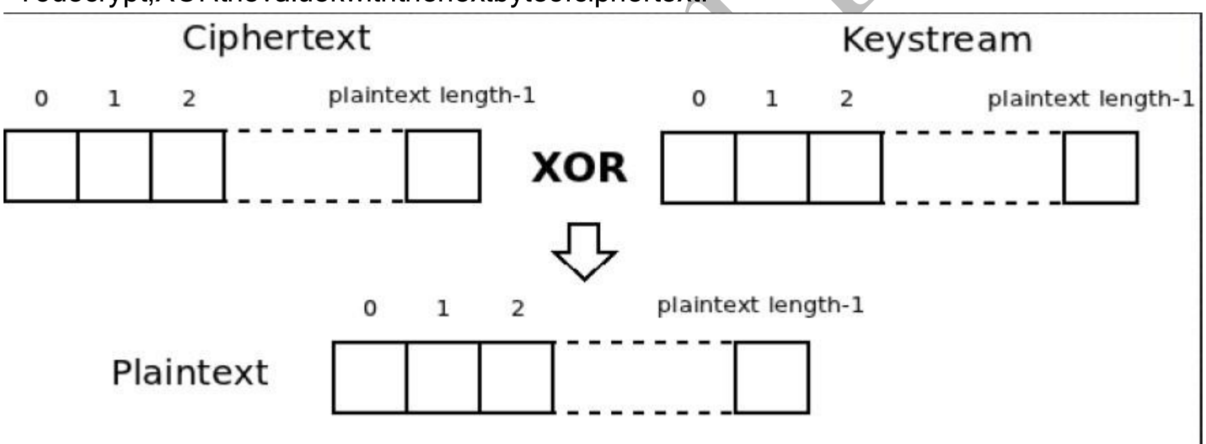


Figure2.35RC4 Decryption

Advantage

- Itisfasterandmoresuitableforstreamingapplication

2.14 KeyDistribution

2.14.1 SymmetricKeyDistributionUsingSymmetricEncryption

- InSymmetrickeyencryption,thetwopartiestoanexchangemustsharethesamekey, and that key must be protected from access by others. Therefore, the term thatrefers to the means of delivering a key to two parties who wish to exchange data,withoutallowingotherstoseethekey.
- For two parties A and B, key distribution can be achieved in a number of ways, asfollows:

1. A canselectakeyandphysically deliverittoB.
2. A thirdpartycanselectthekeyand physically deliverittoAandB.
3. IfAandBhavepreviouslyandrecentlyusedakey,onepartycantransmitthenewkey totheother,encryptedusingtheoldkey.
4. IfAandBeachhasanencryptedconnectiontoathird-partyC,Ccandeliverakeyonthe encryptedlinksto AandB.

- Physical delivery (1 & 2) is simplest - but only applicable when there is personal contactbetween recipient and key issuer. This is fine for link encryption where devices & keysoccurinpairs,butdoesnotscaleasnumberofpartieswhowishtocommunicategrows.3arem ostlybasedon 1or2 occurringfirst.
- A third party, whom all parties trust, can be used as a trusted intermediary to mediate theestablishment of secure communications between them (4). Must trust intermediary not toabuse the knowledge of all session keys.As numbers ofparties grow,some variant of4isonly practicalsolution tothehuge growthinnumberofkeyspotentially needed.

2.14.2 KeyDistributionCentre

- Theuseofakeydistributioncentreisbasedontheuseofahierarchyofkeys.Ataminimum,twolev elsofkeysareused.
- Communicationbetweenend systemsisencrypted usingatemporarykey, oftenreferredtoasaSessionkey.
- Typically,thesessionkeyisusedforthedurationofalogicalconnectionandthendiscarded
- Masterkeyissharedbythekeydistributioncentreandanendsystemoruserandusedtoencryptth esessionkey.

2.14.3 KeyDistributionScenario

- Let us assume that user A wishes to establish a logical connection with B and requires aone-timesessionkey toprotect thedatatransmittedovertheconnection.Ahasamaster key, K_a, known only to itself and the KDC; similarly, B shares the master key $K_b$withthe KDC(Figure2.36).The followingstepsoccur:

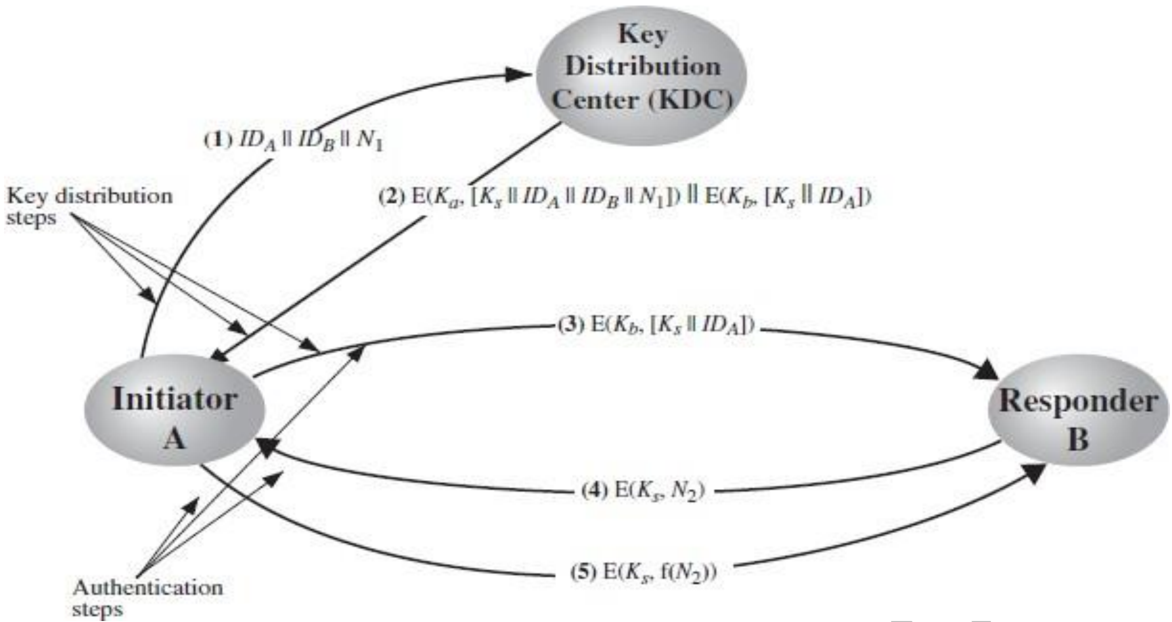


Figure2.36KeyDistributionScenarios

1. An issue a request to the KDC for a session key to protect a logical connection to B. Themessage includes the identity of A and B and a unique identifier, N_1, for this transaction,which we refer to as a nonce. The nonce may be a timestamp, a counter, or a randomnumber; the minimum requirement is that it differs with each request. Also, to preventmasquerade, it should be difficult for an opponent to guess the nonce. Thus, a randomnumberisagoodchoicefora nonce.
2. The KDC responds with a message encrypted using Ka Thus, A is the only one who cansuccessfully read the message, and A knows that it originated at the KDC. The messageincludestwoitemsintendedforA:
   - Theone-timesession key,Ks,tobeusedforthesession
   - The original request message, including the nonce, to enable A to match thisresponsewiththe appropriaterequest

   Thus, A can verify that its original request was not altered before reception by the KDCand, because of the nonce, that this is not a replay of some previous request. Inaddition,themessageincludestwoitems intendedforB:
   - Theone-time session key,Kstobeusedforthesession
   - AnidentifierofA(e.g., its networkaddress),IDA

   These last two items are encrypted with Kb (the master key that the KDC shares with B).They aretobe sentto BtoestablishtheconnectionandproveA'sidentity.

3. A store the session key for use in the upcoming session and forwards to B theinformationthat originated at the KDC for B, namely, E (K_b, [K_s || ID_A]). Because this information isencrypted with K_b, it is protected from eavesdropping. B now knows the session key (K_s),knowsthat the otherparty isA (fromID_A), and knowsthat theinformationoriginatedatthe KDC (because it is encrypted using K_b). At this point, a session key hasbeensecurelydeliveredtoAandB,andtheymaybegintheirprotectedexchange. However,twoadditionalsteps aredesirable:
4. Usingthenewlymintedsessionkeyforencryption,B sendsanonce,N_2,toA.
5. AlsousingK$_s$,Arespondswithf(N_2),wherefisafunctionthatperformssometransformationonN_2 (e.g.,addingone).

2.14.4 SessionKeyLifetime

- The distribution ofsession keys delays the start ofany exchange and places a burdenonnetworkcapacity.Asecuritymanagermusttrytobalancethesecompetingconsiderationsindeterminingthelifetimeofaparticularsessionkey.
- For connection-oriented protocols, one obvious choice is to use the same session keyfor the length of time that the connection is open, using a new session key for eachnew session.
- Ifa logical connection has a very long lifetime, then it would be prudent to changethesessionkeyperiodically,perhapseverytimethePDU(protocoldataunit)sequencenumber cycles.
- Foraconnectionlessprotocol,suchasatransaction-orientedprotocol,thereisnoexplicitconnectioninitiationortermination.
- Thus,itisnotobvioushowoftenoneneedstochangethesessionkey.Themostsecureapproachis to usea new session keyforeach exchange.
- Abetterstrategyistouseagivensessionkeyforacertainfixedperiodonlyorforacertainnumberof transactions.

UNITIIIPUBLICKEYCRYPTOGRAPHY9MATHEMATICSOFASYMMETRICKEYC RYPTOGRAPHY

TOPICS:PUBLICKEYCRYPTOGRAPHYMATHEMATICSOFASYMMETRICKEY CRYPTOGRAPHY: Primes – Primality Testing – Factorization – Euler‘s totient function, Fermat‘sand Euler‘s Theorem - Chinese Remainder Theorem – Exponentiation and logarithmASYMMETRIC KEY CIPHERS: RSA cryptosystem – Key distribution – Key management –Diffie Hellman key exchange - ElGamal cryptosystem – Elliptic curve arithmetic-Elliptic curvecryptography.

PRIMENUMBERS

Primenumbers

Prime numbershavedivisorsof1 andits number itself.

Primefactorisation

To compute **GCD** of any two numbers in prime factorization approach we need tofind**prime factors**ofthetwonumbers.

FermatTheoremorFermat's little theorem

If**a**belongs tointeger,Pisa primenumber that doesnotdivide**a** then a^P congruenta(mod P)ie.,$a^p \equiv a \pmod P$

Inspecialcase

$a^{P-1} \equiv 1 \pmod P$ ifGCD(a,P)=1.wherea andparecoprime.

It is mainly used to solve modular

exponentiation.Eg.Computethevalueof2^{10}mod 11

$2^{10} \equiv 1 \pmod{11}$

Eg.Computethevalueof2^{340}mod11

$(2^{340})=(2^{10})^{34}$ mod 11

$=1^{34}$mod11=>1

//Proof.

Takedivisionalgorithm

a=p.q+r wherecanbe0<=r<=p-1

letg.c.d(a,p) iecoprime

aisnotdivisiblebyphence 1<=r <=p-1

factsays thatif**a** leaves remainderr where1<= r<=p-1on dividingby**p**thenka,1<=k<=p-1alsoleavesremainders from1top-1.

Means
Ifa,2a,3a,…(p-1)asurelygivesremainders1,2,3,…(p-1)
So ifmultiply
a*2a*3a*…*(p-1)a ≡ 1.2.3…(p-1) (mod p)hence a^{p-1}.(p-1)! ≡ (p-1)! (mod p)whichreturns
a^{p-1}≡1(modp)[asmodpcannotdivide(p-1)!]
henceproved.//

Eg.6^{10}mod11

Sol.6^{11-1}mod11
=1[as pertheorem]

Eg.5^{15}mod13
=(5^2mod13)*(5^{13}mod13)
=(25mod13)*5
=(12*5) mod13
=60mod13
=8.

Euler'stheorem

If **n** and **a** are coprime positive intergersthen $a^{phi(n)}$≡1(mod)n
Inthistheoremphi(n)=n-1.
n isprimenumberandphi(n) isEuler'sphifunction.
Euler's phi function is also called Euler's totient function and hence named asEuler'stotienttheoremorEuler'stheorem.

Euler'sphifunctionorEuler'stotientfunction(

Euler's phi function phi (n) returns the numbers of integers from 1 to n,that are relativelyprimeton.
Thephifunctioniscomputedphi(n)using variousmethods.Theyare

1.Ifnisaprime numberthenphi(n)= n-12.Ifnisacompositenumberthen
2.1Findtheprimefactors ofthatnumber andcomputethephifunctionvalueasusedinstep1.otherwise
2.2.Find prime powers(p^n)ofthe given number n. Forcomputingthe phivalue of primepowers wehavetousetheformula
(p^a–p^{a-1}).

Eg. Compute Eulers's totient function for the values 3,81.phi(3)=3-1=2
2.phi(8)=2^3
=2^3- 2^{3-1} (sincep^a –p^{a-1})=2^3 -2^2= 8–4=4

PrimalityTesting Methods

Primalitytestingmethodis a methodtofindandtoprovewhether thegiven numberisprimenumber.

1.NaiveAlgorithm:

NaïveAlgorithmisusedto dividethe giveninputnumberPbyall theintegersstartingfrom2torootofP−1
Ifanyoneofthemis a divisor,thentheinputnumberPis notaprime.Otherwise,itisconsideredas aprimenumber

Algorithm:

1. Pick anyintegerPthat isgreaterthan 2
2. TrytodividePbyallintegers startingfrom2 tothesquarerootofP
3. IfPis divisiblebyanyoneoftheseinteger, wecanconclude thatPisacomposite
4. ElsePisaprimenumber

   Example:
   Findtheprimality testforthe number 100using naïvealgorithm.1.P=100
   2.2,3,4,5,6,7,8,9
   3.Case 1: 100/2 =

   50(composite)Therefore,

   100isnotaprimenumber.

2.Fermat'sPrimalityTest:

IfPis aprimeand**P** does not divide**a,** whichisanaturalnumberthen $a^{P-1} \equiv 1(mod P)$

Example:

Check whether the given number 12 is prime number or not using Fermat'stheorem
GivenP=12

Tocheck whether 12is prime number or not,wehavetocheck $a^{P-1} \equiv 1(mod P)$
$a^{12-1} \equiv 1(mod 12)$
$a^{11} \equiv 1(mod 12)$
Where1<=a<12
Wehavetocalculate $a^{11} mod 12$
If it is equal to 1, then it is called prime number. Otherwise, it is calledcomposite number.
Consider,
a=5 $5^{11} \equiv 1(mod 12)$
(i.e) 5^{11} mod 12=5
Itisnotequalto1
Thereforeitisnotaprime number

3.Miller–RabinPrimalityTest

FunctionMiller-Rabin(x)
w

```
x-1=(2)y        //xistheinput numberforprimality test
                //yis
```

```
select a randomly in the range[2,(x-1)]
Z=a^y mod x
if Z congruent 1(mod x)
then return prime for i=1 to w–1
{
If Z congruent-1(mod x) then return prime
Z=Z^2 mod x
}
return composite
```

Example:
Find the primality for
7x =7
1
Asperalgorithm,x–1=7–1=6=2x3
x =7,w=1,y =3
Z=a^ymod7
a=2(randomly),where [1<=a<=x-1]
Z=2^3mod7=1
Value ofZ=1,7 isconcludedas primenumber

ChineseRemainderTheorem

Statesthatwhenthemoduli ofa systemoflinearcongruenciesarepairwiseprime,thereisauniquesolutionofthesystemmodulo,the productofthe moduli.
x≡a(modm).
ChinesemathematicianSunTsuSuan-Chingaskingthefollowing problem:
‗Therearecertainthingswhosenumberisunknown.Whendividedby3,theremainderis 2;whendividedby5,theremainderis 3; andwhendivided by7,theremainderis2.What willbethenumberofthings?
(Otherwise) Mangos are dividedintogroupsconsisting of3mangos ineachgroupremainingis2.Ifthemangosare dividedintogroupsconsistingof5mangosineachgroupremaining3.
If mangos aredividedintogroupsconsistingof7 mangos ineachgroupremaining 2.Totallyhow manymangosareavailable?

- x≡a1mod(m1)
- x ≡a2mod(m2)
- x ≡a3mod(m3)
- $x=\Sigma(a_iM_iy_i)=(a_1M_1y_1+a_2M_2y_2+a_3M_3y_3) \bmod M$

LetM1,M2,…,Mnbe(pairwise)relativelyprimenumbers.Thenthesystem:Step1:Calculate M

- M=m1*m2*m3… mn.

Step 2:Calculate
M_k=M/m_kStep3:FindInverse
ofM_K(ie)y_k

<u>Findthe Xusing CRT</u>

- x≡2mod(3)
- x≡3mod(5)
- x≡2mod(7)

$a_1=2, a_2=3, a_3=2;\ m_1=3, m_2=5, m_3=7;$

i. $M=m_1 x m_2 x\ m_3=105$.

ii. Foreachequationcalculate

$M_k=M/m_k$(ie)$M_1=M/\ m_1=105/\ 3=35$ M_2 = $M\ /\ m_2 = 105\ /\ 5 = 21$ $M_3=M\ /m_3=105/\ 7= 15$

iii. inverseofM_k**(ie)** y_k

inverseofM_1**(ie)** $y_1=35^{-1}$mod(3)=**35** $^{3-2}$mod (3)=2[sinceFermat'sinverse theorem oreasyinversemethodlike 35x?mod3=1 (ie)2]

$y_2=1; y_3=1$

$X=\Sigma(a_i M_i y_i)=(a_1M_1y_1+a_2M_2y_2+a_3M_3y_3)$modMX=[(2x35x 2)+(21x3x1)+(2x15x1)]mod105

=(140+63+30)mod105

=233 mod105

X=23

Exponentiation

- Exponentiationisatypeofoperationwheretwoelementsareusedinwhichoneelementis considered asabase elementand anotherasanexponentialelement.
- For example,b isan example ofexponential operationwhere xisa base elementandyisanexponentialelement.
- When y is a positive integer, exponentiation is performed in a similar way to repeatedmultiplicationisperformed.
- Modularexponentiationis atypeofexponentiationinwhichamodulodivision operationisperformedafterperformingan exponentiation operation.
- Forexample,**(x^ymod n),**where nisan integernumber.
- The exponentiation is an important concept discussed in many cryptographicalgorithmssuchasRSA,Diffie-Hellman,Elgamal,etc.,

Example:1

11^7mod13

11^2mod13=121mod13=4

11^4mod13=(11^2mod 13x11^2mod13)mod 13

=(4x4)mod13

=16mod13

=3

11^7mod13=(11^4mod13x11^2mod13x11^1mod13)mod13

=(3x4x11)mod13

=(132)mod13

=2

Find the result of 2^{90} mod

13.Solution:

Step 1: Split x and y into smaller parts using exponential rules as shown below:2^{90}mod13=2^{50}x 2^{40}

Step 2:Calculatemodn foreach part

2^{50}mod13=1125899906844

2^{40} mod13=1099511627776 mod 13=3
Step3:Usemodularmultiplicationpropertiestocombinethese2parts,wehave
2^{90} mod13=(2^{50} x 2^{40})mod13
=(2^{50}mod13x2^{40}mod13) mod13
=(4x 3)mod13=(12) mod13=12

LogarithmsorIndices

- Discrete logarithms are logarithms defined with regard to multiplicative cyclicgroups. If*G*is amultiplicativecyclicgroupand*g*isa generator of*G*,thenfromthedefinitionofcyclicgroups,weknoweveryelement*h* in *G* canbe writtenas g^x for some*x*. The discretelogarithmto thebase*g*of*h* in the group*G* isdefined tobe*x* .
- For example,ifthegroupisZ_5^*,andthegenerator is2,thenthe discretelogarithmof1 is4because2^4≡1mod5.
- Input:p-prime number,a-primitiverootofp,b-a residuemod p.
- Goal:Findk suchthata^k = b(mod p). (In other words,findthepositionofyin thelargelistof{a,a^2,. .., a^{q-1}}.
- 14isa primitiveroot of19.
- ForexampleL$_{14}$(5)=10 mod19,because14^{10}=5(mod19).
- the inverse problem to exponentiation is to find the **discrete logarithm** of anumbermodulop
- thatistofindxwherea=b^xmodp
- writtenasx=log$_a$bmodp orx=ind$_{a,p}$(b)
- if a isaprimitive rootthenalwaysexists

ASYMMETRIC KEY

CIPHERSPUBLICKEYCRYPTOGRAPHY:

Principlesofpublickeycryptosystems

Theconceptofpublickeycryptographyevolvedfromanattempt toattacktwoofthemostdifficult problems associated with symmetric encryption. Key distribution undersymmetrickeyencryptionrequireseither
(1) Two communicantsalreadyshare akey,whichsomeonehasbeen distributedtothem
(2) Theuseofakeydistributioncenter.
- Digitalsignatures.

CharacteristicsofPublickeycryptosystems

Publickeyalgorithmsrelyononekeyforencryptionandadifferentbutrelatedkeyfordecryption.Thesealgorithms havethefollowingimportant characteristics:

- It is computationally infeasible to determine the decryption key given only theknowledgeofthecryptographic algorithm andtheencryptionkey.

In addition, some algorithms,suchasRSA,alsoexhibitthefollowingcharacteristic:

- Either ofthetworelatedkeyscanbeusedforencryption, withtheotherusedfordecryption.

INGREDIANTSOFPUBLICKEYCRYPTOGRAPHY

1. Plaintext:Thisis thereadablemessageordatathatis fedintothealgorithmasinput.
2. Encryption algorithm: The encryption algorithm performs various transformations on theplaintext.
3. **Public and private keys:** This is a pair of keys that have been selected so that if one is usedfor encryption, the other is used for decryption. The exact transformations performed by thealgorithmdependonthepublicorprivatekey thatisprovidedasinput.
4. Ciphertext:Thisisthescrambledmessageproducedasoutput. Itdependsontheplaintextandthekey. For agivenmessage,twodifferentkeys willproducetwodifferent ciphertexts.
5. Decryption algorithm: This algorithm accepts the ciphertext and the matching key andproducestheoriginalplaintext.

Encryption:

Theessential stepsarethe following:

1. Eachusergenerates apairofkeystobeusedforencryptionanddecryptionof messages.
2. Each user places oneofthetwokeys inapublic register or other accessiblefile.This isthepublickey. Thecompanionkeyiskeptprivate.
3. IfAwishestosendaconfidentialmessagetoB,Aencrypts themessageusingB"s public key.
4. WhenBreceivesthemessage,itdecrypts usingits privatekey.

With this approach(Fig), all participants have access to public keys and private keys are generatedlocallybyeachparticipantandtherefore, neednotbedistributed.

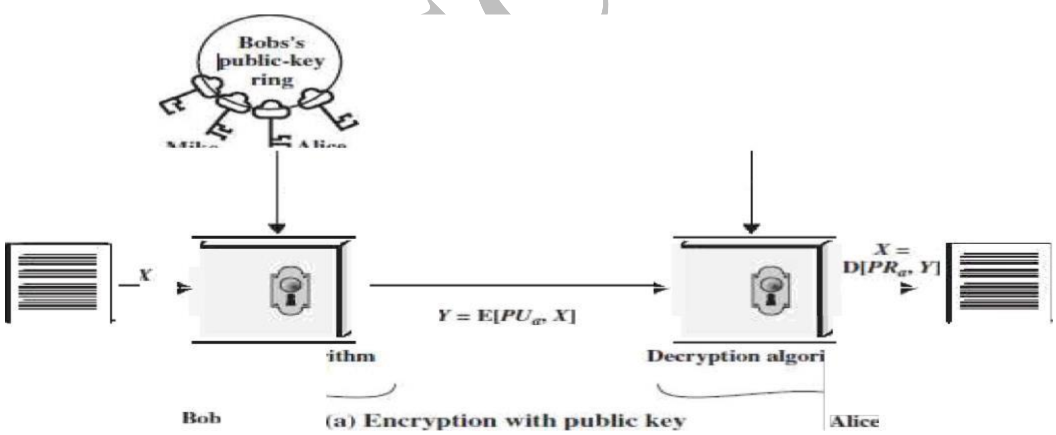


Fig. Public KeyCryptographyFor Authentication

Let theplaintextbeX=[X1, X2.X3,..,X]where misthenumber oflettersinsomefinitealphabets. SupposeAwishestosendamessagetoB.

Bgeneratesapairof keys:apublickeyKUbandaprivatekeyKRbKRbisknownonlytoB,whereasKUbispubliclyavailablea ndthereforeaccessiblebyA.

With themessageXandencryptionkeyKUb as input,A forms thecipher textY=[Y1,Y2, Y3›

. Yn]•

i.e.,Y=EKUb(X)

ThereceivercandecryptitusingtheprivatekeyKR_b i.e.,$X=D_{KRb}(Y)$

Theotherapproach(usingsender¯sprivatekeyforencryptionandsender¯spublickeyfordecryption)willprovide authentication whichisillustratedinthe followingdiagram(Fig 2.26).

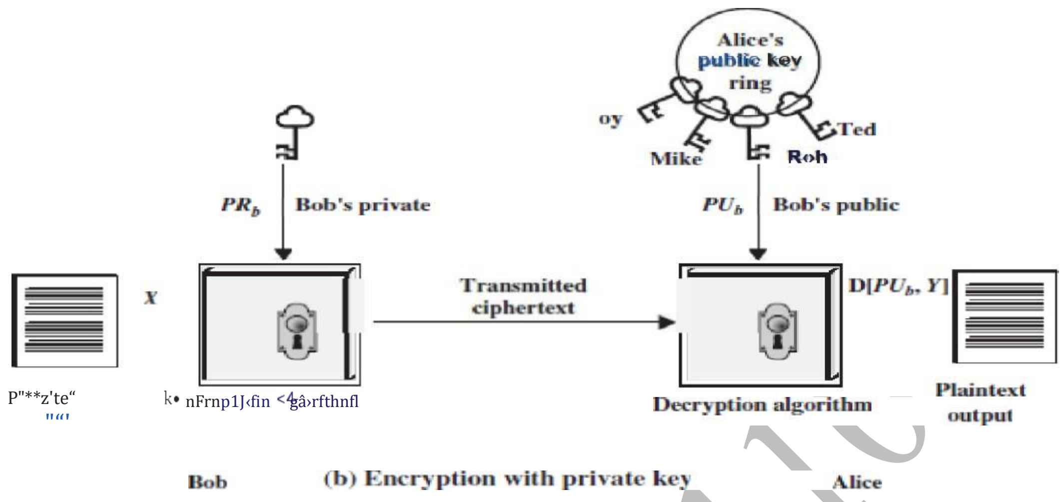


Fig.PrivateKeyCryptographyFor Authentication

The encryptedmessageservesasadigitalsignature. Itis important toemphasizethattheencryptionprocessjust describeddoesnotprovideconfidentiality.

Differencesbetween publickeyEncryption and conventionalEncryption

Conventional Encryption	Public-Key Encryption
Needed to Work:	*Needed to Work:*
.Thesame-algorithmu'irh thesame kcvis usedfor enczygtiunand dccption. *. Thesender andreceivermustshareh algorilhrn andthekey. /'Vce‹Jed *fnrSeciJriiy.-* 1. Thelk€'YfTtMStbfikCpt 5tCF£:I. 2.Trrnusl beirnpnssihlcorat lczstimpractical‹+ decipheramessageilthekeyi‹keptsecret. 3.Kriov•'1cJyc 3fthealgorithm pJus sample:of cipherrc xtmum beinsufficienttodv\crrninc thekey.	1.OnouJ**urilhmir**usedfor«ncmplionanJarcJared alyurirhmfordcc'ptionallhapair of I:c\'.unc*inr* tncptiunandonefordecryption. *.Thesenderandru-cei›crmustcachhaveoneofthe matchedpcir oikcTz(n3tlhcsame unc). ' "¯_‖'"!-' 1.Oneof the la'O keys mustbekeptsecret. *.JImustbeimpossibleorall«astimpracticalr‹1 decipheramessageif ‹oneoflhckcYsiskz'j•rsccrcl. 3.Knowledge‹›f the algori\kmplusuncoirhckc}z plus sampler ofciphertextmeltbeirlsufficicnlm determinetheothe rkey'.

Public KeyCryptographyfor Security

ThereissomesourceAthatproducesamessage in plaintext,*X*—— [X1,+2,.,X].Theelements ofX arelettersinsomefinitealphabet.

ThemessageisintendedfordestinationB.Bgeneratesarelatedpairofkeys:apublickey,*PUT,*and aprivatekey,*PRb*PRtisknownonlytoB,whereasPL/hispubliclyavailableandthereforeaccessiblebyA.W iththemessageandtheencryptionkeyPL/,asinput, AformstheciphertextY=[Y1.*Yo..,YN]*

Y=E(PU_B,X)

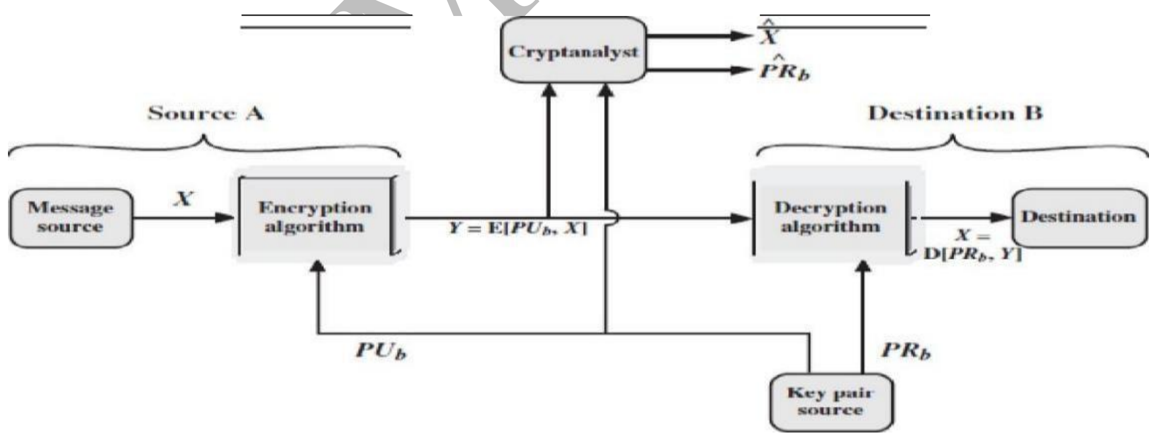


Figure.Public-KeyCryptosystem:Secrecy

The intended receiver, in possession of the matching private key, is able to invertthetransformation:

X——D(PRd›

An adversary, observing *Y* and having access to *PUT,* but not having access to *PRbor X,*must attempt to recover X and/or *PRb* It is assumed that the adversary does have knowledge oftheencryption(E)anddecryption(D)algorithms.

If the adversary is interested only in this particular message, then the focus of effort is torecover A by generating a plaintext estimate X *. Often, however, the adversary is interested inbeing able to read future messages as well, in which case an attempt is made to recover PRtbygeneratinganestimate*PRb.*

$$Y = E(PR_{a1})X =$$
$$D(PU_{a1}Y)$$

In this case, A prepares a message to B and encrypts it using A‘s private key beforetransmitting it. B can decrypt the message using A‘s public key. Because the message wasencryptedusingA‘s privatekey,onlyAcouldhavepreparedthemessage.

Therefore, the entire encrypted message serves as **a digital signature.** In addition, it isimpossibletoalterthemessagewithoutaccesstoA‘sprivatekey,sothemessageisauthenticatedbothintermsofsource andintermsofdataintegrity.

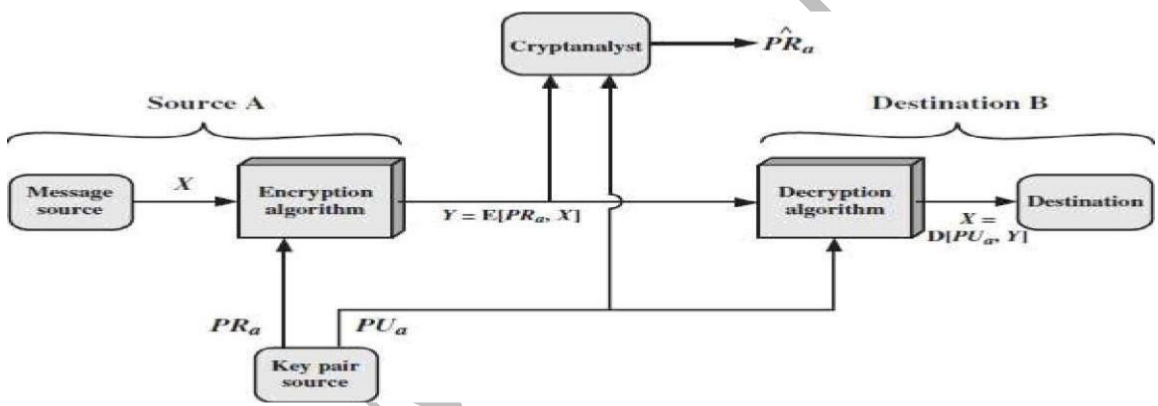


Figure.Public-KeyCryptosystem:Authentication

It is important to emphasize that the encryption process depicted in above Figuresdoesnot provide confidentiality. That is, the message being sent is safe from alteration but not fromeaves dropping. This is obvious in the case of a signature based on a portion of the message,becausetherestofthemessageistransmitted intheclear.Eveninthecaseofcompleteencryption, as shown in Figure13, there is no protection of confidentiality because any observercandecryptthemessagebyusingthesender‘spublickey.

AuthenticationandSecrecy

It is, however, possible to provide both the authentication function and confidentiality by adoubleuseofthepublic-keyscheme(Figure2.29):

Ciphertext Z=EKUb[EKRa(X)]

PlaintextX=EKUa[EKRb(Y)]

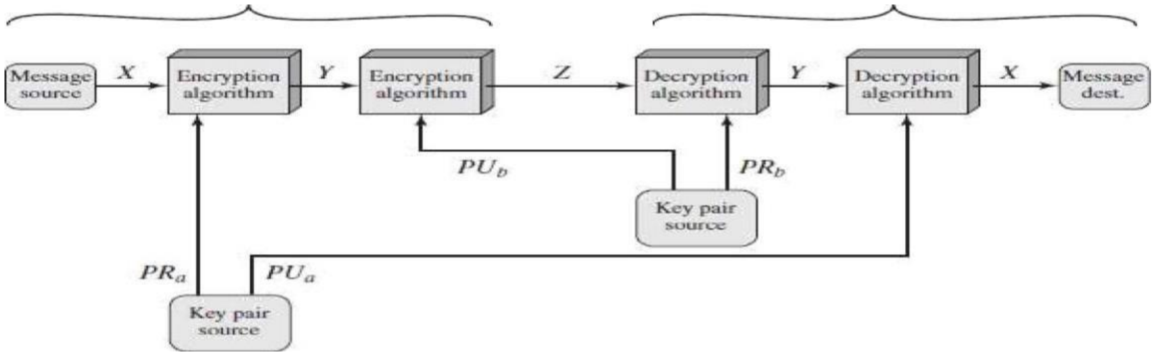


Figure.Public-KeyCryptosystem:Authenticationand**Secrecy**

Initially, the message is encrypted using the sender¯s private key. This provides the digitalsignature. Next, we encrypt again, using the receiver¯s public key. The final ciphertext can bedecryptedonlybytheintendedreceiver,whoalonehasthematchingprivatekey.Thusconfidentialityisp rovided.

ApplicationsforPublic-KeyCryptosystems

Wecanclassifytheuseof public-keycryptosystemsintothreecategories

1. Encryption/decryption:Thesenderencryptsamessagewiththerecipient'spublickey.
2. Digital signature: The sender ¯signs" a message with its private key. Signing is achievedby a cryptographic algorithm applied to the message or to a small block of data that is afunctionofthemessage.
3. Key exchange: Two sides cooperate to exchange a session key. Several differentapproaches arepossible,involvingtheprivatekey(s)ofoneor bothparties.

Requirementsforpublickeycryptography

- It iscomputationallyeasy foraparty Btogeneratea pair[KUb,KRb]
- It iscomputationallyeasyforasenderA,knowingthepublickeyandthemessagetobeencrypted M, togeneratethecorrespondingciphertext:C=EKUb(M).
- Itis computationallyeasyfor thereceiverBtodecrypttheresultingciphertextusingtheprivatekeytorecovertheoriginal message:

M=DKRb(C)=DKRb[EKUb(M)]

- It is computationally infeasible for an opponent, knowing the public key KUb, todeterminetheprivatekeyKRb
- It is computationally infeasible for an opponent, knowing the public key KUb, and aciphertext C, torecovertheoriginal messageM.
- The encryptionand decryptionfunctions can beapplied ineither order:

M=EKUb[DKRb(M)=DKUb[EKRb(M)]

Public-KeyCryptanalysis

Attack Type 1 :

Thepublic-key encryptionschemeis vulnerabletoabrute-force attack; therefore uselargekey.Thetradeoffisthat makesuseofsomesortofinvertiblemathematicalfunction.

Thereforechoosekeysizesuchthatthebruteforceattackisnotpossible,atthesametimeshouldnotbetooslow forgeneraluse.

Attack type2:

Attackis ofother types(i.e.) giventhe algorithmandthepublickeydeduceprivate key.Thismethodhasnotbeensuccessfultilldate.

Attack Type 3:

A probable-message attack. When a confidential message is to be transmitted usingDES,theattackerwillfindall2^{56}possiblekeysusingthepublickeyanddiscovertheencrypted keybymatchingthegeneratedciphertextandtheactualcipher.Thisattackcanbeavoidedbyappendingsomerandom bitstothemessage.

RSAALGORITHM

ItwasdevelopedbyRivest,ShamirandAdleman.Thisalgorithmmakesuseofanexpression with exponentials. Plaintext is encrypted in blocks, with each block having a binaryvaluelessthansomenumbern.

The RSA scheme is a cipher in which the plaintext and cipher text are integers between 0andn- 1for somen. Atypicalsizefornis1024bits, or309decimaldigits.Thatis,nislessthan 2^{1024}

Thatis,theblocksizemustbelessthanorequaltolog2(n);inpractice,theblocksizeis k-b its, where2^k<n<2^{k1}*.Encryptionanddecryptionareofthefollowingform,forsomeplaintext block MandciphertextblockC:

$$c = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Boththesenderandreceiverknowthevalueofn.thesenderknowsthevalueofeandonly the receiver knows the value of d. thus, this is a public key encryption algorithm with a publickey of KU = {e, n) and a private key of KR = {d, n}. For this algorithm to be satisfactory for publickeyencryption,thefollowingrequirementsmustbemet:

1. It ispossibletofindvaluesofe,d,nsuchthatM^{ed}=Mmodn forallM<n.
2. ItisrelativelyeasytocalculateM^{e}andC^{d}forallvaluesofM<n.
3. Itis infeasibletodeterminedgiven eandn.

Select p, q	p and q both prime, $p \neq q$
Caena•• ml•› F—!› i•J— 1)	
Scloci intcgcrc	$\gcd(\phi(n), e) = 1; 1 < e < \phi(n)$
Calculared	$d \equiv e^{-1} \pmod{\phi(n)}$
Public key	$PU = \{e, n\}$
Private key	$PR = \{d, n\}$

Encryption by Bob with Alice's Public Key	
Plaintext:	$M < n$
Ciphertext:	$C = M^e \bmod n$

Decryption by Alice with Alice's Public Key	
Ciphertext:	C
Plaintext:	$M = C^d \bmod n$

Fig.TheRSAAlgorithm

Let usfocusonthefirst requirement.Weneedtofindtherelationshipoftheform:

$$M^{ed} = M \bmod n$$

Giventwoprimenumberspandqandtwointegers, nandm,suchthatn=pqand0<m<n,andarbitraryintegerk, thefollowingrelationshipholds

k6(n)+1 k(p-1)(q-1)+1 mmodn

where6(n)—Eulertotient function,whichisthenumberofpositiveintegerslessthannandrelativelyprimeton.wecanachievethedesiredrelationship,if

ed=k6(n)+1

Thisisequivalenttosaying:

ed-1 mod 6(n)

d=e-1mod6(n)

Thatis,eanddaremultiplicativeinversesmod6(n).Accordingtotheruleofmodulararithmetic,this is trueonlyifd(andthereforee) isrelativelyprimeto6(n).Equivalently, gcd(6(n),d)= 1.

Wearenow readytostatetheRSAscheme.Theingredientsarethefollowing:

*p,q,*twoprimenumbers	(private,chosen)
n —— *pq*	(public,calculated)
e,withgcd(6*(n),*e)=1;1<e<6*(n)*	(public, chosen)
d-Ke-1(mod6(n))	(private,

calculated)ThestepsinvolvedinRSAalgorithm forgeneratingthekeyare

- Selecttwoprimenumbers, p=17andq=11.

- Calculaten=p*q=17*11=187Calculate6(n)
- = (p-1)(q-1)= 16*10= 160.
- Selectesuch thateisrelatively primeto6(n)=160andless than6(n);wechoose

e = 7.

- Determine *d* such that *de* K 1 (mod 160) and *d* < 160.The correct value is *d* —— 23,because 23 • 7 = 161 = (1 • 160) + 1; *d* can be calculated using the extended Euclid'salgorithm

Theresulting keysare publickey*PU*——{7,187}andprivatekey*PR*—— {23,187}.

Theexample showstheuseofthesekeysfor a plaintextinputof*M*—— 88.Forencryption,weneedtocalculateC =887mod187.

Exploitingthe propertiesofmodulararithmetic, wecan dothisasfollows.

88^{7}mod187 =[(88^{4}mod187)•(88^{2}mod 187)x(88^{1}mod 187)]mod187
88^{1}mod187=88
88^{2}mod187 =7744 mod187=77
88^{4}mod187 =59,969,536mod 187 =132
88^{7}mod187 =(88 •77 • 132) mod 187 =894,432mod187=11

Fordecryption,wecalculate*M*—— 1123mod187:

11^{23}mod187=[(11^{1}mod187) •(11^{2}mod187)•(11^{4}mod187)•(11' mod187)•
(11 mod187)]mod 187
11mod187=11
^{1} mod187=121
11^{8}mod 187=14,641mod187=55
11^{4}mod187 =214,358,881mod 187=33
11^{23}mod187=(11•121•55 •33•33)mod187=79,720,245mod187=88

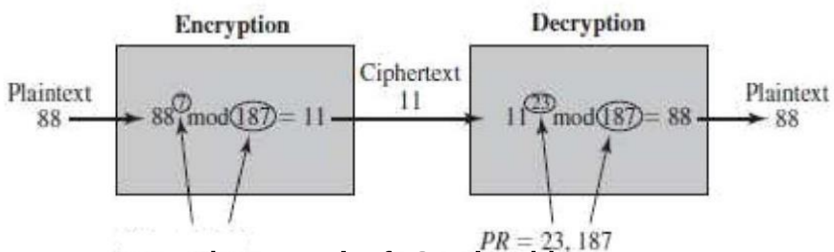


Fig.ExampleofRSAAlgorithm

SecurityofRSA:

Thereare threeapproachestoattackthe RSA:

1. Bruteforce:This involves tryingallpossibleprivatekeys.
2. Mathematical attacks: There are several approaches, all equivalent in effort tofactoringthe productoftwoprimes.
3. Timingattacks: Thesedependontherunningtimeofthedecryptionalgorithm.

Type1RSAAttack: DefensetoBruteForceattack:

Uselargekeyspace(i.e)largenumberofbitsineanddthebettersecuredbut problemsare,

1. Increasescomputing power
2. FactoringProblem

Type

2RSAAttack:MathematicalAttack:Mathematical

approachtakes3forms:

- Factorn=p*q,hencefind 6(n)andthend.
- Determine 6(n) directlywithout determiningpand qandfind d.d=e-1(mode(n))
- Find ddirectly,without firstdetermination6(n).

Type3RSAAttack: Timingattacks:

This attackislearningfor2reasons

1. Comes completelyfromunexpected direction
2. Cipher textonlyattack

Attack:

If the system does lastly the modular multiplication in majority of cases but takes longertimeinfew cases.Theaverageisalsolonger.

Theattackisdonebit

bybitStartwithleftmostbit

b„

Supposefirstjbits areknown.

For agivencipher texttheattacker completesthejiteration.

Ifthebitissetthend<-(d*a)modn.

MethodstoovercomeTimingattacks:

1. Constant exponentiation time: All exponentiations take the same amount of time beforereturningaresult. Thisisasimplefix butdoesdegradeperformance.
2. Random delay: Better performance could be achieved by adding a random delay to theexponentiationalgorithmtoconfusethetimingattack.
3. Blinding: Multiply the cipher text by a random number before performing exponentiation.This process prevents the attacker from knowing what cipher text bits are being processedinside the computer and therefore prevents the bit-by-bit analysis essential to the timingattack.

KEYMANAGEMENT

Therearetwousesofpublickeycryptographyregardingtheissuesof keydistribution.Theyare

1. Distributionofpublickeys
2. Useofpublickeyencryptiontodistributesecret keys

DistributionofPublicKeys

Several techniques have been proposed for the distribution of public keys. Virtually alltheseproposalscanbegroupedintothefollowing generalschemes:

a) Public announcement
b) Publiclyavailabledirectory
c) Public-keyauthority
d) Public-keycertificates

(a) PublicAnnouncementofPublicKeys

Inpublic-keyencryptionthepublickeyispublic.Thus,ifthereissomebroadlyaccepted public-key algorithm, such as RSA, any participant can send his or her public key toanyotherparticipantor broadcastthekeytothecommunity atlargeasshowninFigure2.32.

Figure.UncontrolledPublic-KeyDistribution

Disadvantage:

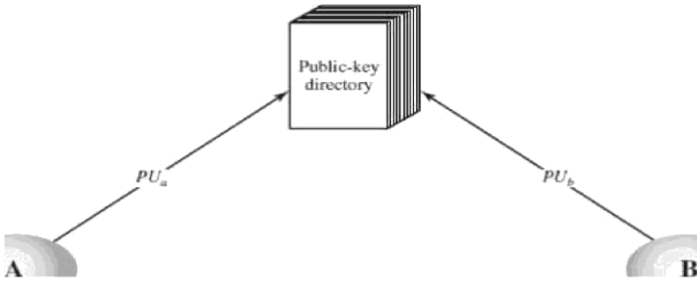
Anyonecanforgesuchapublicannouncement.Thatis,someusercouldpretendtobeuserAa nd senda publickeytoanotherparticipantorbroadcastsuch apublic key.

Until such time as user A discovers the forgery and alerts other participants, the forgeris able to read all encrypted messages intended for A and can use the forged keys forauthentication.

(b) PubliclyAvailableDirectory

A greater degree ofsecurity can be achieved by maintaining a publicly availabledynamic directory of public keys. Maintenance and distribution of the public directory wouldhave to be the responsibility of some trusted entity or organization as shown in Figure 2.33.Suchascheme wouldincludethe followingelements:

1. Theauthoritymaintainsadirectorywitha(name,publickey}entryforeachparticipant.
2. Each participant registers a public key with the directory authority. Registration wouldhavetobeinpersonorby some formofsecure authenticated communication.
3. A participantmay replace the existing key with a new one atany time,due to eitherthe key has been used for a large amount ofdata, or the corresponding private keyhasbeencompromised insome way.
4. Participants could also access the directory electronically. For this purpose, secure,authenticated communication from theauthoritytotheparticipantismandatory



FigurePublic-KeyPublications

Vulnerabilities:

- 6Tampertherecords ofpublickeydirectories.
- 6 If an adversary succeeds in obtaining or computing the private key of the directory authority,the adversary could authoritatively pass out counterfeit public keys and impersonate anyparticipant andeavesdroponmessagessenttoanyparticipant.

(c) Public-KeyAuthority

Stronger security for public-key distribution can be achieved by providing tighter controlover the distribution of public keys from the directory. A typical scenario is illustrated in Figure2.34.

As before, the scenario assumes that a central authority maintains a dynamic directory ofpublic keys of all participants. In addition, each participant reliably knows a public key for theauthority, with only the authority knowing the corresponding private key. The following steps(matchedbynumbertoFigure2.34)occur:

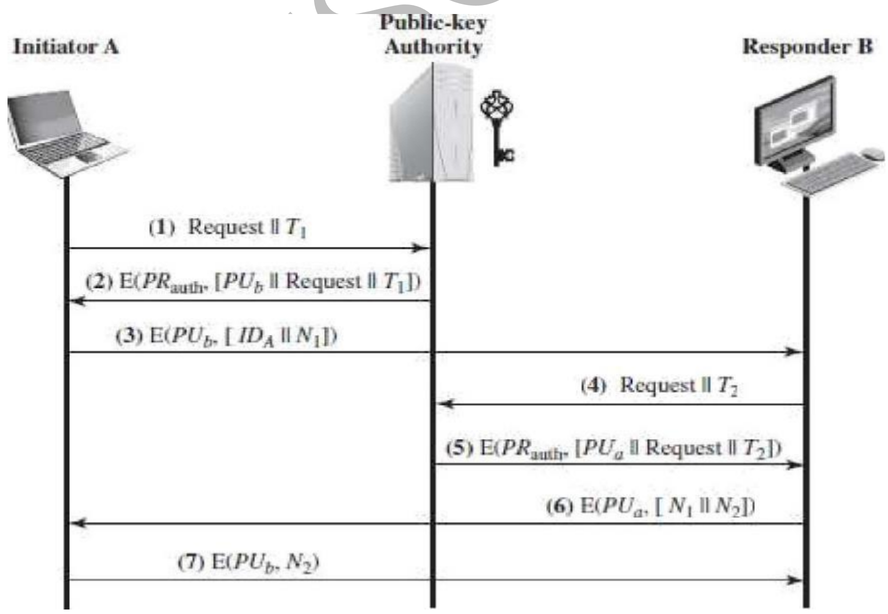


Figure.Public-KeyDistributionScenario

1. A sends a time stamped message to the public-key authority containing a request for thecurrentpublickey ofB.

2. The authority responds with a message that is encrypted using the authority's private key,*PRauth*Thus, Ais abletodecryptthemessageusingtheauthority'spublickey.Therefore, Aisassuredthat themessageoriginatedwiththeauthority. Themessageincludesthefollowing:

- B's publickey, PL/,whichA canuseto encrypt messages destinedforB
- Theoriginalrequest,to enableAto matchthisresponsewiththe correspondingearlier request and to verify that the original request was not altered before receptionbytheauthority
- The original timestamp, so A can determine that this is not an old message from theauthority containinga keyotherthanB's current publickey

3. A stores B's public key and also uses it to encrypt a message to B containing an identifierofA *(IDA)*andanonce(/\/1),whichisusedtoidentify this transactionuniquely.

4,5 B retrieves A's public key from the authority in the same manner as A retrieved B'spublickey.

6.BsendsamessagetoAencryptedwithKaandcontainingA'snonce(/\/1)aswellasanew nonce generated by B (/\/2) Because only B could have decrypted message (3), thepresenceof/\/1in message(6)assures A thatthecorrespondentisB.

7. Areturns2. encryptedusingB'spublickey,toassure Bthatitscorrespondentis A.

Thus,a total ofseven messages are required. However,the initial fourmessagesneed be used only infrequently because both A and B can save the other's public key forfutureuse,a techniqueknownascaching.

Disadvantages:

- Bottleneck attheauthority.

(d) Public-KeyCertificates

The scenario of Figure 2.35 is attractive, yet it has some **drawbacks.** The public-keyauthority could be somewhat of a bottleneck in the system, for a user must appeal to theauthority for a public key for every other user that it wishes to contact. As before, the directoryofnamesandpublic keysmaintainedbytheauthority isvulnerabletotampering.

An alternative approach is to use **certificates** that can be used by participants toexchange keyswithoutcontactinga public-key authority.

A certificate consists of a public key plus an identifier of the key owner, with the wholeblocksigned by atrusted third party.

A user can present his or her public key to the authority in a secure manner, and obtain acertificate. The user can then publish the certificate. Anyone needed this user's public key canobtainthecertificateandverifythat it isvalidbywayoftheattachedtrustedsignature.

1. Any participantcanreadacertificate todeterminethenameandpublic keyofthecertificate'sowner.

2. Any participant can verify that the certificate originated from the certificate authority and is notcounterfeit.
3. Onlythecertificateauthoritycancreateandupdatecertificates.
These requirements are satisfied by the original proposal in. Denning added the followingadditionalrequirement:
4. Anyparticipantcanverifythecurrencyofthecertificate.

A certificate scheme is illustrated in Figure. Each participant applies to the certificateauthority,supplyingapublickeyandrequestingacertificate.

(a) Obtaining certificates from CA

Public-Keydistributionof SecretKeys usingpublic keycryptography:

- Usepreviousmethodstoobtainpublic-key
- Canuseforsecrecy orauthentication
- Public-keyalgorithmsareslow sousuallywanttouseprivate-keyencryptiontoprotectmessagecontents, Henceneedasessionkey

a) Simple
b) Secretkeydistributionwith confidentialityand authentication
c) Hybrid
d) Diffie Hellmankeyexchange

(a) SimpleSecretKeyDistribution:

1. Agenerates apublic/privatekeypair(KUa,KRa) andtransmits amessagetoB consistingofKUaandanidentifierof A,IDA.
2. Bgeneratesasecretkey ,and transmits itto A,encrypted withA‘spublickey.
3. AcomputesDKRa[EKUa[s]]torecoverthesecret key.
BecauseonlyAcandecryptthemessage, onlyAandBwill know theidentityofKs-
4. A discardsKUaandKRaand BdiscardsKUa•

Advantages:

- Nokeysexistbeforethestart ofthecommunicationnokeyexistafterthecompletionofcommunication
- Securefromeavesdropping

Disadvatages:

- Replayattack
- Meet inthemiddleattack
- Ageneratesapublic/privatekeypair*{PUa,PRaj* andtransmits amessageintendedforBconsistingofPL/aandanidentifier ofA,*IDA.*
- Dinterceptsthemessage,createsitsownpublic/privatekeypair*PUd,PRO*andtransmi ts*PUs*00*IDA* toB.
- Bgeneratesasecretkey,*Ks,*andtransmitsE(PL/s,*Ks).*
- Dinterceptsthemessageandlearns *Ks*bycomputingD(PRd,E(PL/d,*Ks)).*
- DtransmitsE(PL/a, *Ks)*toA.

(b) SecretKeyDistributionwithConfidentialityand Authentication:

1. A usesB'spublickeyto encrypt a messagetoB containingan identifier ofA (IDA)andanonce(N1),whichisusedtoidentifythistransactionuniquely.
2. B sendsamessageto Aencrypted withKUaand containingA'sdecryptedmessage (1) ,thepresenceofN1inmessage(2)assuresAthatcorrespondentisB.
3. A returns Ne. encryptedusingB's publickey, to assurer Bthat its correspondent is A.

4. A select a secret key s find sends M = EKUb[EKRa[Ks]] TO B. Encryption of this messagewith B's public key ensures that only B can read it.;encryption with A's private key ensuresthatonlyAcouldhavesentit.
5. ComputesDxUa[DKRb[M]]torecoverthesecretkey.

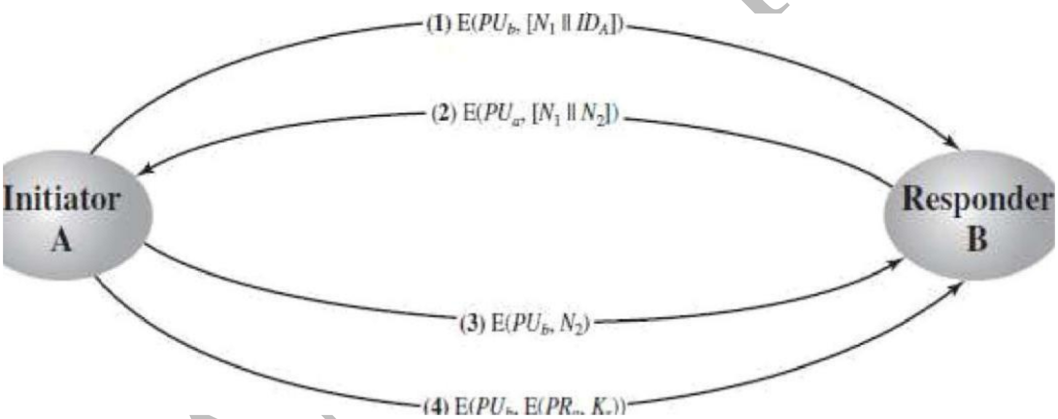


Figure.PublicKeyDistributionofsecretKeys

Advantages:

Scheme ensures bothconfidentialityandauthenticationin theexchangeofa secretkey.

(c) AHybridScheme

Public-key scheme isused to distribute the master keys. The following rationale is providedfor usingthisthree-levelapproach:

1. Performance:

Thepublickeyencryptionisusedoccasionallytoupdatethemasterkeybetweenusesand KDC

Whenthedistributionofsessionkeysis donebypublickeyencryptiontheperformancedegradesbecauseofhighcomputationneededbyP.K.E.

2. **Backward compatibility:** The hybrid scheme is easily overlaid on an existingKDCschemewithminimaldisruptionorsoftwarechanges.

The additionof apublic-keylayerprovidesa secure,efficientmeans ofdistributingmasterkeys.

DIFFIEHELLMANKEYEXCHANGE

The purpose of the algorithm is to enable two users to exchange a key securely that canthenbeusedforsubsequentencryptionofmessages.TheDiffie-Hellmanalgorithmdependsforits effectivenessonthedifficultyofcomputingdiscretelogarithms.

First, we define a primitive root of a prime number p as one whose power generate all theintegers from1to(p-1)i.e.,if_a'isaprimitiveroot ofaprimenumber p,thenthenumbers

$$a \bmod p, a^2 \bmod p, \ldots a^{p-1} \bmod p$$

aredistinct andconsists ofintegers from1to(p-1) insomepermutation.

Forany integer_b'anda primitiveroot_a'ofa primenumber_p',wecan findauniqueexponent_i'such that

$$b \equiv a^i \bmod p \text{ where } 0 \le i \le (p-1)$$

The exponent _i'is referred toasdiscretelogarithm.

<u>TheAlgorithm</u>

Figure 2.37 summarizes the Diffie-Hellman key exchange algorithm.There are publiclyknown numbers: a prime number _q' and an integer a that is primitive root of q. suppose users Aand B wish to exchange a key. User A selects a random integer X_A‹ q and computes $Y_A = a^{X_A} \bmod q$.

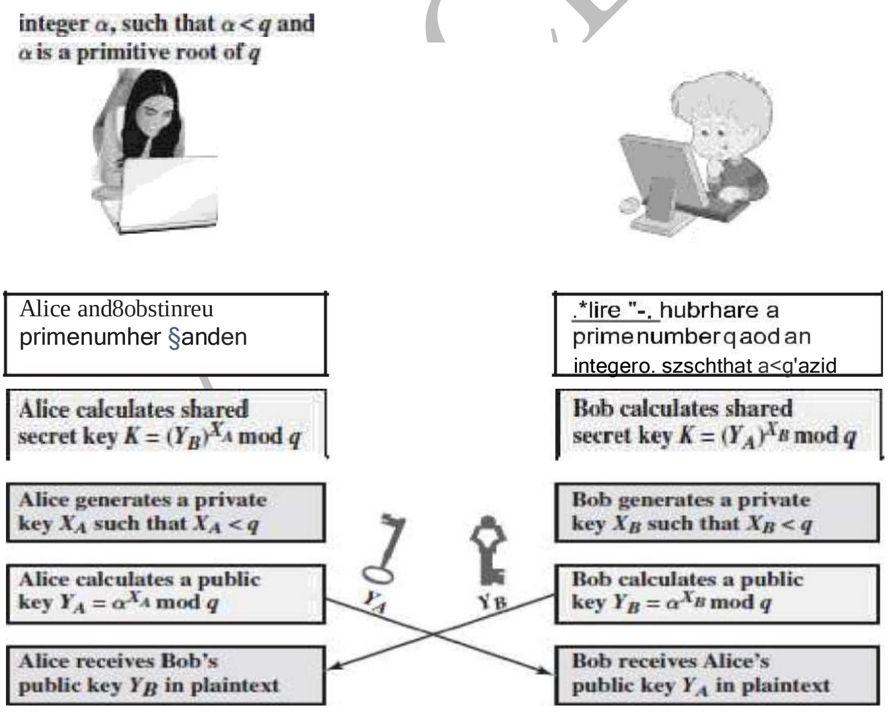


Fig.DiffieHellmanKeyExchange

Similarly, user B independently selects a random integer X_B< q and computes $Y_B = a^{X_B} \bmod q$. Each side keeps the X value private and makes the Y value available publicly to the otherside.

UserAcomputesthe key as

K=(YB)modqand

UserB computesthekeyas

$K=(YA)^{XB}modq$

Thesetwocalculations produceidenticalresults.

K=(YB) modq

$=(a^{XB} modq) modq$

$=(a^{XB})mod q$

$—(a^{XB}) modq$

$=(a^{XB} modq) modq$

$=(Y_A)^{XB} modq$

GlobalPublic Elements	
q	primenumber
α	a<qandaistheprimitiverootofq
UserAKeygeneration	
Select XAXy< q	
Calculate publicYrYA modq	
UserBKeygeneration	
Select XBXB‹q	
Calculate publicYB $YB=^{XB}modq$	
Generationofsecretkey	
byUserAK=(YB)modq	
Generationofsecretkey byUserB	
$K = (Y_A)^{XB}mod\ q$	

Theresultisthattwosideshaveexchangedasecretkey.Thesecurityofthealgorithm lies in the fact that, while it is relatively easy to calculate exponentials modulo aprime,itisverydifficulttocalculatediscretelogarithms.

KeyExchangeProtocols

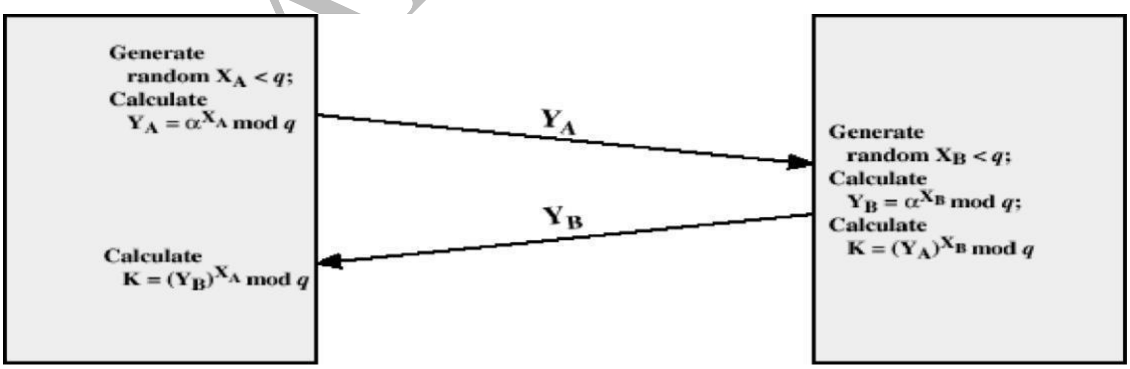


Figure.Diffe-Hellman KeyExchange

The protocol depicted in figure 2.38 is insecure against a man-in-the-middle attack. Suppose AliceandBobwishtoexchangekeys, andDarthis theadversary. Theattackproceedsas follows:

1. Darthprepares fortheattackbygeneratingtworandomprivatekeysXD1andXD2andthencomputingthecorrespondingpublickeysYD1and
Yo2-2. Alicetransmits YAtoBob.
3. Darthintercepts YAandtransmitsYD1toBob. Darthalsocalculates = $(YA)^X$Dmod q.
4. BobreceivesYD1andcalculatesK1=(YD1)Bmodq.
5. Bob transmitsXA to Alice.
6. Darthintercepts XAandtransmits YD2toAlice.Darthcalculates1"$(YB)^X$D1mod q.
7. AlicereceivesYo2and calculates 2'$(YD2)^X$modq.

At this point, Bob and Alice think that they share a secret key, but instead Bob and Darth sharesecret key K1 and Alice and Darth share secret key All future communication between Bob andAliceiscompromisedinthefollowingway:

1. AlicesendsanencryptedmessageM: E(2.M).
2. Darthintercepts theencryptedmessageanddecryptsit,torecover M.
3. DarthsendsBobE(K1.M)or E(K1.M'), whereM' isanymessage

Example:

Key exchange is based on the use of the prime number *q* —— 353 and a primitive root of 353, inthiscasea=3.AandBselectsecretkeys*EA*——97and *XB*——233,respectively.

Each computes its publickey:
Acomputes*YA*3^7 mod353=40.
Bcomputes*YB*——3^{233}mod353= 248.
Aftertheyexchangepublickeys,eachcancompute
thecommonsecretkey:Acomputes*K*—— *(Yy)*mod353= 248 mod353= 160.
Bcomputes*K*——$(Y A)^{XB}$ mod353=40 ^{23}*mod353=160.

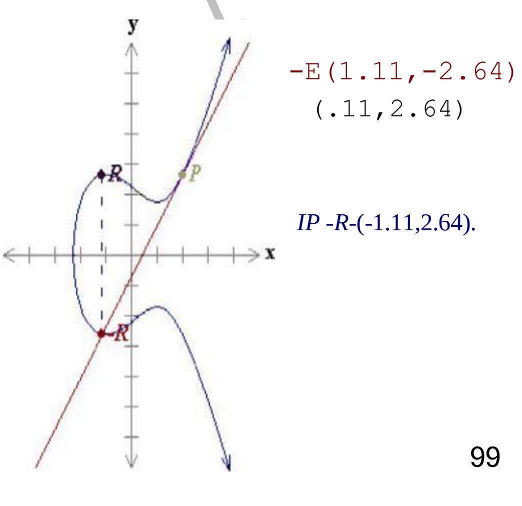
ELLIPTICCURVEARITHMETIC

EllipticCurves:

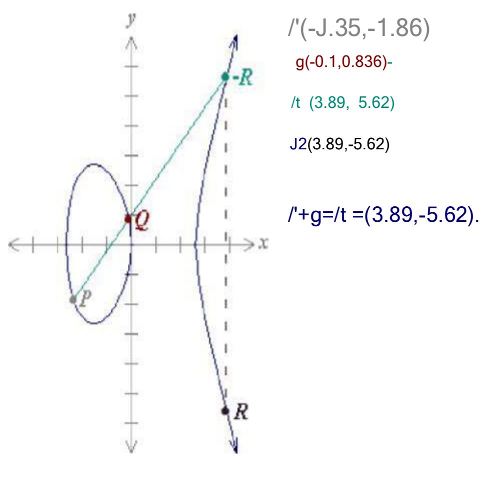
An ellipticCurve is aCubic equation oftheform

$$Y^2+axy+by=x^3+cx^2+dx+e$$

Wherea,b,c,dandearerealnumbers



$y^2=x^3-3x+5$



$y^2=x^3-7x$

Aspecial addition operation isdefined overellipticcurves and with the inclusion of a point ¯0"calledpointatinfinity.

Ifthreepointsareonalineintersectinganellipticcurve,thentheirsumisequaltothispoint atinfinityO (whichacts astheidentityelementforthis additionoperation)

EllipticCurves overGaloisfield:

AnellipticgroupovertheGaloisFieldEd(a,b)isobtainedbycomputingx^3 +ax+bmodp for 0ñ xñ p. The constants a&b are non-negative integers smaller than the prime number p mustsatisfythecondition.

$$4a^3+27b^2 \text{ modp} 10$$

For eachvalueofx, oneneedstodeterminewhetheror notit isaquadraticresidue. Ifnotthenthepointis notintheellipticgroupEd(a,b)

Additionandmultiplicationoperationoverellipticgroups:

LetthepointsP=(x1.y1)andQ =(X,Y2)beintheellipticgroupEd(a,b)andO bethepointatinfinity.

TherulesforadditionovertheellipticgroupEd(a,b)are:
1. P+O=O+P=P

2. Ifx2=x1andy2= -y1,thatisP=(x1,y1)andQ=(X2,Y2)'(x1.-y1)'-P ThenP+Q= O

3. IfQ I -P,thentheirsumP+Q =(x3.y3)isgivenby

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

and

$$\lambda = \begin{cases} \frac{y2—+j}{x_2 - x_1} & p, \quad ,;; \\ \frac{3x + a}{2J,} & if \quad P = Q \end{cases}$$

EllipticCurveEncryption:

Ellipticcurvecryptographycanbeusedtoencrypttheplaintext messageM,intociphertext. The plain text message M is encoded into a point PM from the finite set of points in theelliptic group,Ed(a,b).

Thefirst stepconsistsinchoosingageneratorpoint, G cEd(a, b), suchthatthesmallestvalueofnforwhichnG=oisaverylargeprimenumber.

Theelliptic groupEd(a,b)andthegeneratorpoint G are madepublic.

Eachuserselectaprivatekey,nA<nandcomputethepublickeyPAasPA=nAG

ToencryptthemessagepointPMforBob(B),
Alice(A) chosesarandomintegerkandcomputetheciphertext pairofpoints c

UsingBob's publickeyPB

Pc' [(KG),(PM B)]

After receiving the ciphertext pair of points, Pc. Bob multiplies the firstpoint,(KG) with his privatekeynBandthenaddstheresult tothesecondpointintheciphertextpair ofpoints(PM+ KPB)

(PM+ B)–[nB(KG)]=(PM+KnBG)-[nB(KG)]=PM

whichis theplaintext point,correspondingtotheplaintext messageM.

OnlyBobknowingtheprivatekeynB. canremovenB(KG) fromthesecondpointoftheciphertextpair ofpoint, i.e(PM +KGB), andhenceretrievetheplaintextinformationPM

<u>Ellipticcurvecryptography</u>

SecurityofECC:

1. The cryptographic strength of elliptic curve encryption lies in the difficulty for a crypt analysttodeterminethesecret randomnumberkfromKP&Pitself.

2. Thefastestmethodtosolvethis problem(knownaselliptic curvelogarithmproblemis thepollardfactorizationmethod).

3. The computational complexity for breaking the elliptic curve cryptosystem, using the pollardmethodis3.3x 10^{10}MIPSyearsfor anellipticcurvekeysizeofonly150bits.

4. ForcomparisonthefastestmethodtobreakRSA,usingGeneralNumberFieldSievemethodto factor the composite integer n in to the two prime p & q requires 2x10^{11} MIPS years for a 768bit RSAkey&3x10^{11}MIPSyearsforaRSAkeylength1024
5. IftheRSAkeylengthisincreasedto2048bits,theGNESmethodwillneed3x10^{20}MIPS yearstofactornwhereasincreasingtheellipticcurvekeylengthtoonly24bitswillimposea computational complexityof1.6x10_2MIPS years.

AnalogofDiffie-HellmanKeyExchange:

Keyexchangeusingelliptic curvescanbedonein thefollowingmanner.

Firstpickalargeinteger q, whichiseither aprimenumber por aninteger oftheform2‖ andelliptic curveparameters aandb.This definestheelliptic groupofpoints Ed(a,b).

Next, pickabasepointG=(x1.y1)inEd(a,b)whoseorderisaverylargevaluen.Theorder nofapoint G onanelliptic curveis thesmallestpositiveinteger nsuchthatnG =O.Eq(a,b) andG areparametersofthecryptosystemknowntoallparticipants.

1. AselectsanintegernAlessthann. ThisisA'sprivatekey. AthengeneratesapublickeyPAnAxG; thepublickeyisapointinEd(a, b)•
2. Bsimilarlyselectsaprivatekey nBandcomputesapublickey PB

3. AgeneratesthesecretkeyK= nAxPB BgeneratesthesecretkeyK=nB xEA.

GlobalPublicelements	
Ed(a,b) an	Elliptic curvewithparameters a,bandq, where qis aprimeor integeroftheform2‖
G	pointonellipticcurvewhoseorderislargevaluen
UserAKeyGeneration	
Select private	**nAnA<n**
CalculatepublicPA	PAnAX G
UserBKey Generation	
Select private	nBnB‹n
CalculatepublicPA	PB=nBX G
Calculationofsecretkey byUserA	
K=nAXPB	
CalculationofsecretkeybyUserB	
K=nBXPA	

Figure.ECCDiffie-Hellman KeyExchange

UNITIVMESSAGE AUTHENTICATIONANDINTEGRITY

UNITIVMESSAGEAUTHENTICATIONANDINTEGRITYAuthenticationrequirement-Authentication function – MAC – Hash function – Security of hash function and MAC – SHA – Digitalsignatureandauthenticationprotocols-DSS-EntityAuthentication:Biometrics,Passwords,ChallengeResponseprotocols-Authenticationapplications-Kerberos,X.509

4. AUTHENTICATIONREQUIREMENT

Communicationacrossthenetwork, thefollowingattackscanbeidentified.

Disclosure–releaseofmessagecontentstoanypersonorprocessnotpossessingtheappropriatecryptographickey.

Trafficanalysis-discoveryofthepatternoftrafficbetweenparties.

- In a connection oriented application, the frequency and duration of connections could bedetermined.
- In either a connection oriented or connectionless environment, the number and length ofmessagesbetweenpartiescouldbedetermined.

Masquerade–insertionofmessagesintothenetworkfromfraudulentsource.Thiscanbecreationofmessagebytheattackerusingtheauthorizedport.

Contentmodification–changestothecontentsofamessage,includinginsertion,deletion,transposition,andmodification.

Sequencemodification–anymodificationtoasequenceofmessagesbetweenparties,includinginsertion,deletion,andreordering.

Timingmodification-delayorreplayofmessages.

- Inaconnectionorientedapplication,anentiresessionorsequenceofmessagescouldbe replay of some previous valid session, or individual messages in the sequence couldbedelayedorreplayed.
- Inaconnectionlessapplication,an individualmessagecouldbe delayedorreplayed.

Sourcerepudiation-denialoftransmissionofmessagebysource.

Destination repudiation-denialofreceiptofmessagebydestination.

4.1. AUTHENTICATIONFUNCTION

Anymessageauthenticationordigitalsignaturemechanismcanbeviewedashavingfundamentally twolevels.

Atthelowerlevel,theremustbesomesortoffunctionthatproduces anauthenticator,avaluetobeusedtoauthenticateamessage.

At the higher-level, low-level function is then used as primitive in a higher-level authenticationprotocolthatenablesareceivertoverifytheauthenticityofamessage.

Thetypesoffunction thatmay beused toproduceanauthenticatoraregroupedinto threeclasses.

MessageEncryption-theciphertextoftheentiremessageservesasitsauthenticator.

MessageAuthenticationCode(MAC)-apublicfunctionofthemessageandasecretkeythatproducesafixedlengthvaluethatservesastheauthenticator.

HashFunction-apublicfunctionthatmapsamessageofanylengthintoafixed-lengthhashvalue,whichservesastheauthenticator.

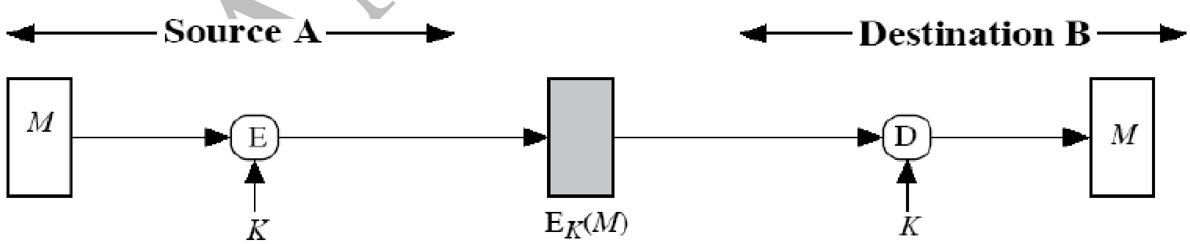
MessageEncryption:

Message encryption Message encryption by itself can provide a measure of

authentication.Theanalysisdiffersfromsymmetricandpublickeyencryptionschemes.

(a) Ifsymmetricencryption(fig.a)isused then:

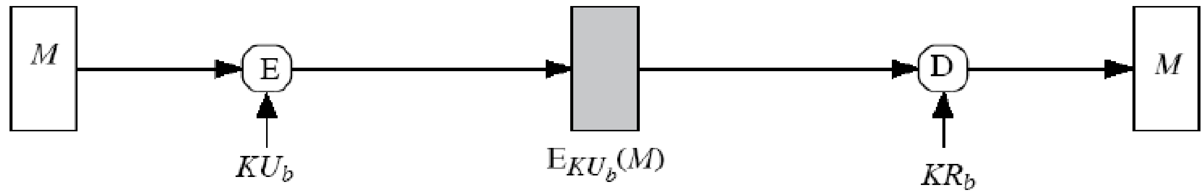
- A messagem, transmitted from source A todestinationBisencryptedusingasecretkeysharedby AandB.
- Sinceonly senderand receiverknows keyused
- Receiverknowssendermusthavecreatedit.Henceauthenticationisprovided.
- Knowcontentcannothavebeenaltered.Hence confidentialityisalsoprovided.
- Ifmessagehas suitable structure,redundancyorachecksumtodetectanychanges
- ThereforeSymmetricEncryptionprovidesauthenticationandconfidentiality.



(a).Symmetrickeyencryptionconfidentiality,authenticationandsignature

(b) Ifpublic-keyencryption(Figb)isused:

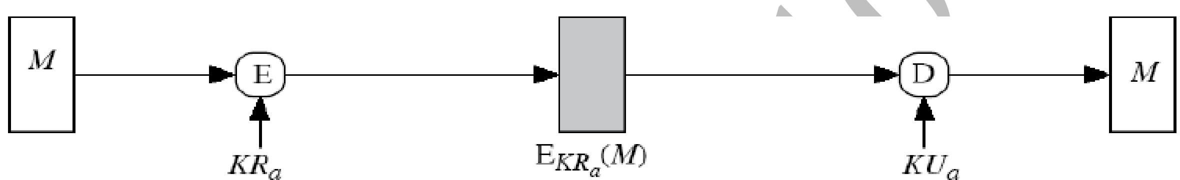
This method is the use ofpublic key cryptography which provides confidentiality only.The sender A makes use of the public key of the receiver to encrypt the message. Here there isno authentication because any user can use B‟s public key to send a message and claim thatonly Ahassentit.



(b) Publickeyencryptionconfidentiality

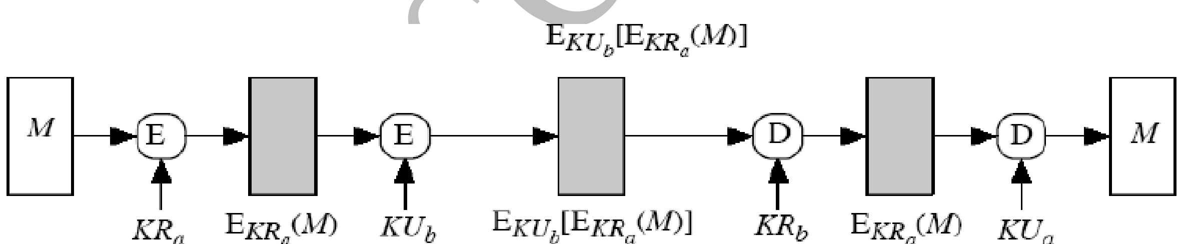
Inthis method(Figc) to have only authentication,the message isencrypted with the sender‟sA‟sprivatekey.ThereceiverBusesthesender‟sA‟spublickeytodecryptthemessage.NowA cannot deny that it has not transmitted since it only knows its private key. This is called asauthenticationorDigital Signature.Hencetheproblemisthe,

- Receivercannotdetermine whetherthepacketdecrypted containssomeusefulmessageorrandombits.
- Theproblemisthatanyonecandecryptthe messagewhentheyknow thepublickey ofsenderA.



Figure(c) Publickeyencryptionauthentication andsignature

Thismethod(Figd)providesauthentication,confidentialityanddigitalsignature.Buttheproblem with this method is the complex public key cryptography algorithm should be appliedtwiceduringencryptionandtwiceduringdecryption.



Figure(d)Publickeyencryption confidentiality,authenticationandsignature

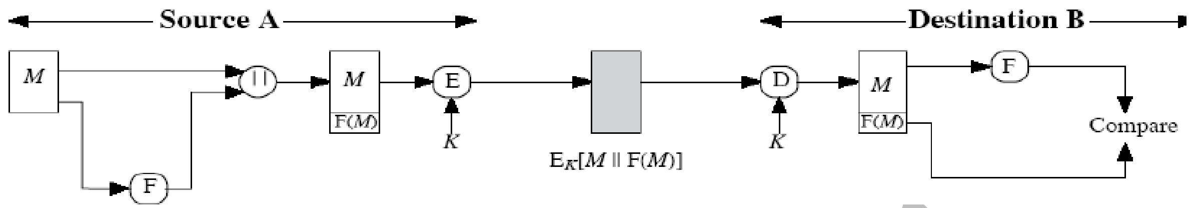
Supposethemessagecanbeanyarbitrarybitpattern,inthatcase,thereisnowaytodetermine automatically, at the destination whether an incoming message is the ciphertext of alegitimatemessage.

One solution to this problem is to force the plaintext to have some structure that is easilyrecognizedbutthatcannotbereplicated withoutrecourseto theencryptionfunction.

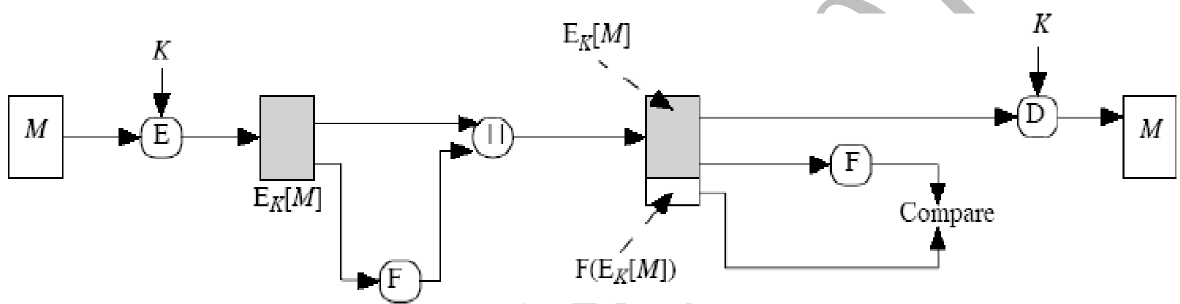
Appendanerrordetectingcode,alsoknownasFrameCheckSequence(FCS)orchecksum to each message before encryption „A‟ prepares a plaintext message M and thenprovides this as input to a function F that produces an FCS. The FCS is appended to M and theentireblockisthenencrypted.

At the destination, B decrypts the incoming block and treats the result as a message withanappendedFCS.BappliesthesamefunctionF to attempttoreproduce theFCS.

If the calculated FCS is equal to the incoming FCS, then the message is consideredauthentic. *In the internal error control, the function F is applied to the plaintext, whereas inexternalerrorcontrol,Fisappliedtotheciphertext(encryptedmessagefigeandd).*



(e) Internal errorcontrol



(f) Externalerrorcontrol

4.2.MAC

An alternative authentication technique involvesthe use ofsecret key to generate asmallfixedsizeblockofdata,knownascryptographicchecksumorMACthatisappendedtothemessag e.

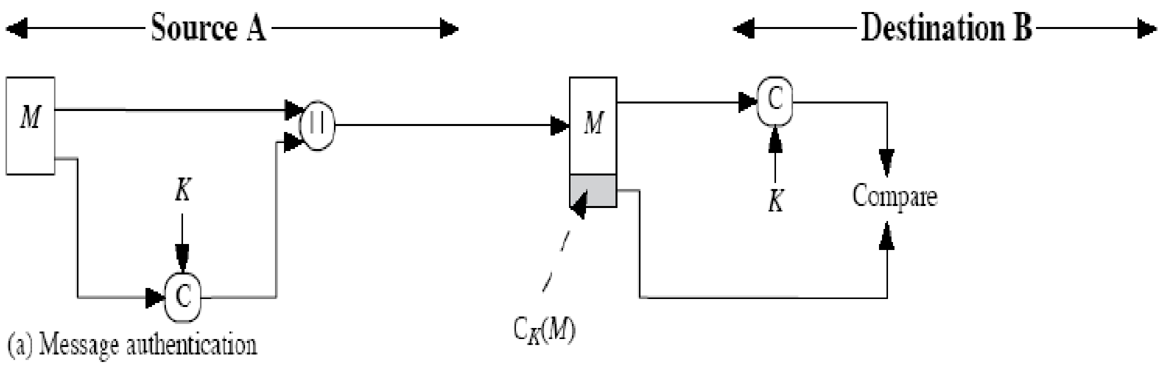
This technique assumes that two communication parties say A and B, share a commonsecret key „k". When A has to send a message to B, it calculates the MAC as a function of themessage andthe key.

$$MAC = C_K(M)$$

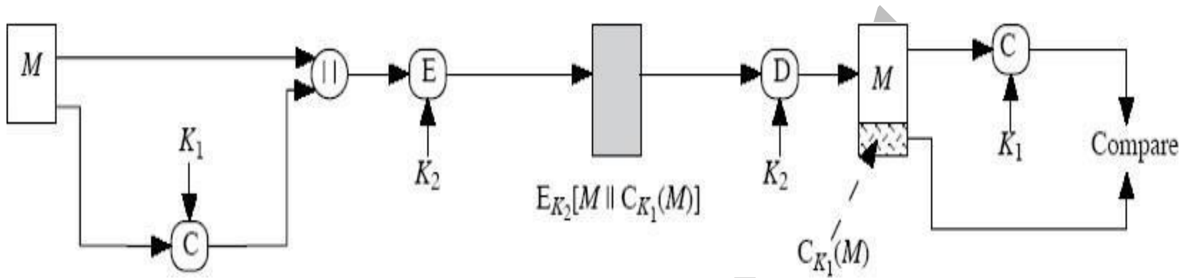
WhereM-inputmessage C-MACfunction K – Shared secret

keyThemessage plus MAC aretransmittedtotheintended recipient. Therecipientperforms thesamecalculationonthereceivedmessage,usingthesharedsecretkey,togenerateanew MAC.

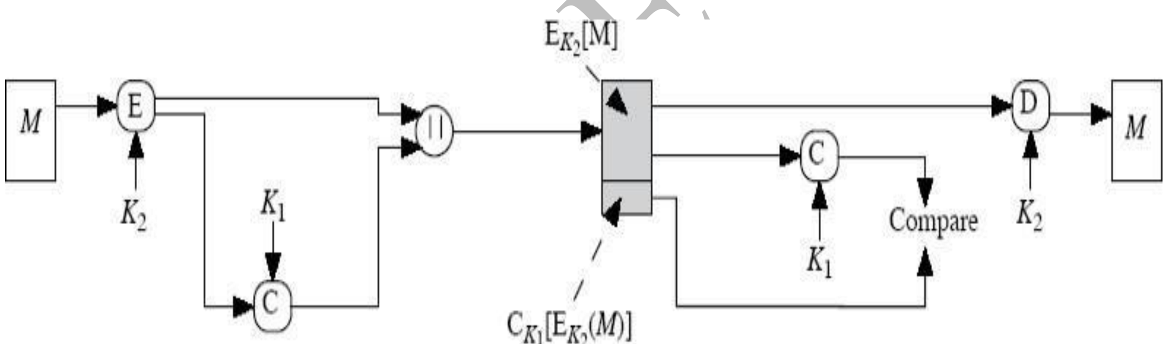
The received MAC is compared to the calculated MAC. If it is equal, then the message isconsidered authentic(Fig g and h). A MAC function is similar to encryption. One difference isthat MAC algorithm need not be reversible, as it must for decryption. In general, the MACfunctionisamany-to-onefunction.



(g) MessageAuthentication



Figure(h)Messageauthenticationandconfidentiality, authenticationtiedtoplaintext



Figure(i)Messageauthenticationandconfidentiality,authenticationtiedtociphertextRequirem entsforMAC:

Whenanentiremessageisencryptedforconfidentiality,usingeithersymmetricorasymmetric encryption, the security of the scheme generally depends on the bit length of thekey.

Barringsomeweaknessinthealgorithm,the opponentmustresortto abrute-forceattack using all possible keys. On average, such an attack will require 2(k-1) attempts for a k-bitkey.

If confidentiality is not employed, the opponent has access to plaintext messages andtheirassociatedMACs.Suppose k>n;thatis,supposethatthe keysizeisgreaterthanthe

MAC size. Then, given a known M_1 and MAC_1, with $MAC_1 = C_K(M_1)$, the cryptanalyst canperformMAC_i=CK_i(M1)forallpossiblekey valuesKi.

Atleastone keyis guaranteedtoproduceamatchofMAC_i=MAC_1.

Note that a total of 2^k MACs will be produced, but there are only $2^n < 2^k$ different MAC values.Thus, a number of keys will produce the correct MAC and the opponent has no way of knowingthe correct key. On average, a total of$2^k/2^n = 2^{(k-n)}$ keyswill produce a match. Thus, theopponentmustiterate theattack:

Round1

Given:M_1,$MAC_1 = C_K(M_1)$
ComputeMAC_i=$C_{Ki}(M_1)$forall2^kkeysNum
berofmatches≈$2^{(k-n)}$

Round2

Given:M_2,$MAC_2 = C_K(M_2)$
ComputeMAC_i=$C_{Ki}(M_2)$forthe$2^{(k-n)}$keysresultingfrom
Round1Numberofmatches≈$2^{(k-2xn)}$ andsoon

ConsiderthefollowingMACalgorithm.LetM= $(X_1||X_2||...||X_m)$bea messagethatistreatedas aconcatenationof64-bitblocksX_i. Thendefine

$$\Delta(M)=X_1+X_2 \ldots X_m C_k$$
$$(M)=E_k(\Delta(M))$$

Thus, thekeylengthis56bitsandtheMAC lengthis64bits. Ifanopponentobserves {M||C(K,M)},abrute-force attempt to determineKwillrequireat least 2^{56}encryptions.

ButtheopponentcanattackthesystembyreplacingX1through $X_{m-1}$with anydesiredvaluesY_1throughY_{m-1}andreplacing$X_m$with $Y_m$where$Y_m$iscalculated as follows:

$$Y_m=Y_1+Y_2 \ldots\ldots Y_{m1}+\Delta(M)$$

Theopponentcannowconcatenatethenewmessage,whichconsistsofY_1throughY_m,withthe original MAC to form a message that will be accepted as authentic by the receiver. With thistactic,anymessageoflength$64X_{(m-1)}$bitscanbefraudulentlyinserted.

Then the MAC function should satisfy the following requirements: The MAC function shouldhavethe followingproperties:

- If an opponent observes M and $C_K(M)$, it should be computationally infeasible for theopponenttoconstructamessageM‟suchthatC_K(M‟)=C_K(M)
- C_K(M)shouldbeuniformlydistributedinthesensethatforrandomlychosenmessages,M and M‟, the probability that C_K(M) = C_K(M‟) is 2^{-n} where n is the number of bits in theMAC.
- LetM‟beequaltosomeknowntransformationon M.i.e.,M‟=f(M).

MACbased on DES

One of the mostwidely used MACs,referred to as Data Authentication Algorithm (DAA) isbasedonDES.

The algorithm(Fig 2) can be defined as using cipher block chaining (CBC) mode of operation ofDESwithaninitializationvectorofzero.Thedatatobeauthenticatedaregroupedintocontiguous 64-bit blocks: D_1, D_2... D_n. if necessary, the final block is padded on the right withzeros to form a full 64-bit block. Using the DES encryption algorithm and a secret key, a dataauthenticationcode(DAC)iscalculated asfollows:

$O_1=E_K(D_1)$
$O_2 = E_K(D_2 +$
$O_1)O_3$
$=E_K(D_3+O_2)$
......
$O_N=E_K(D_N+O_{N-1})$

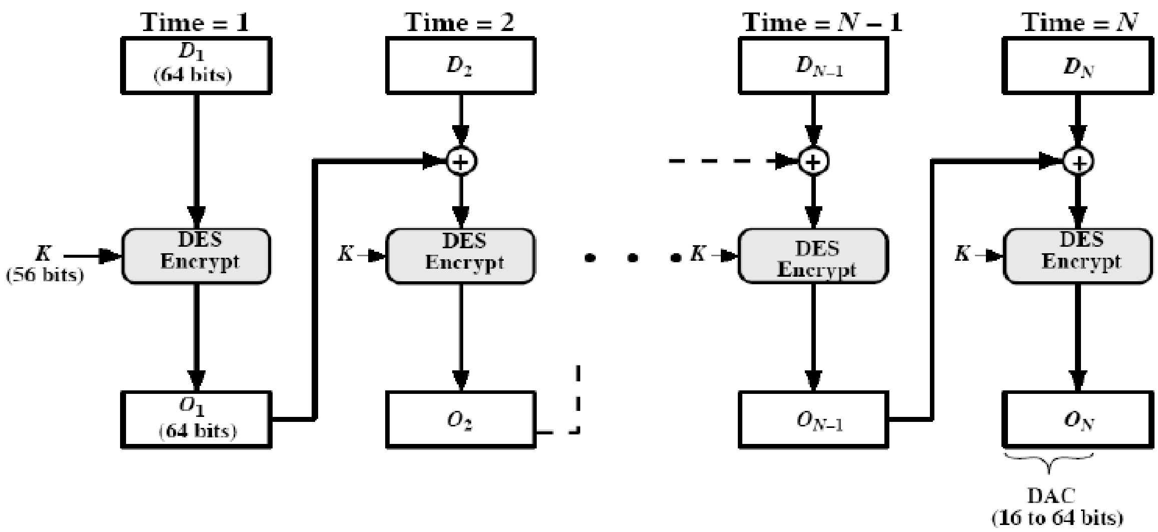


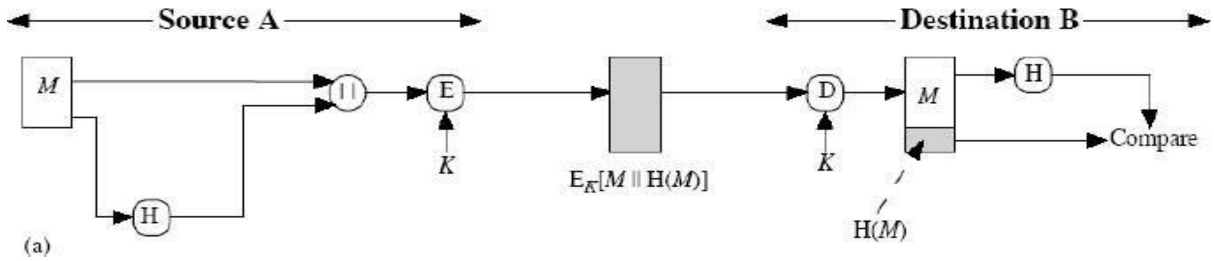
Figure.2 Data Authentication

Algorithm4.3.HASHFUNCTION

A variation on the message authentication code is the one way hash function. As withMAC, a hash function accepts a variable size message M as input and produces a fixed-sizeoutput,referred to as hashcodeH(M).

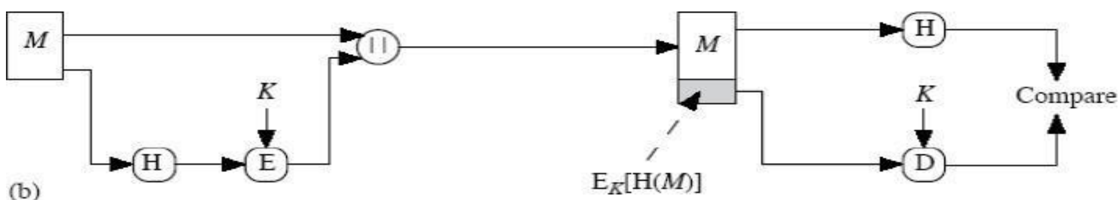
Unlike a MAC, a hash code does not use a key but is a function only of the inputmessage. The hash code is also referred to as a message digest or hash value. There arevarieties of ways in which a hash code can be used to provide message authentication, asfollows:

In Fig (a) The message plus the hash code is encrypted using symmetric encryption. This isidentical to that of internal error control strategy. Because encryption is applied to the entiremessageplusthehash code,confidentiality isalsoprovided.

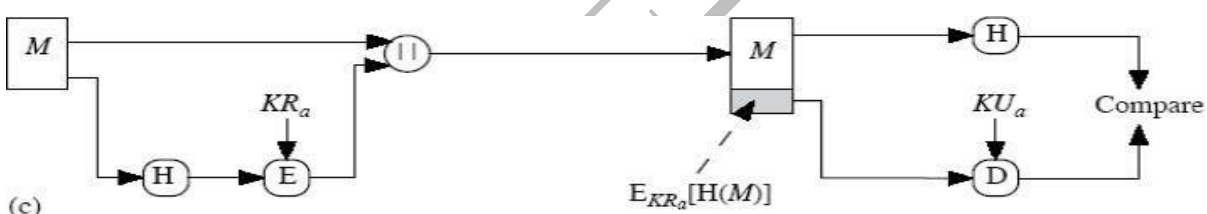


Figure(a)HashFunction

In Fig(b) Only the hash code is encrypted, using symmetric encryption. This reduces theprocessingburden forthoseapplications thatdonotrequireconfidentiality.

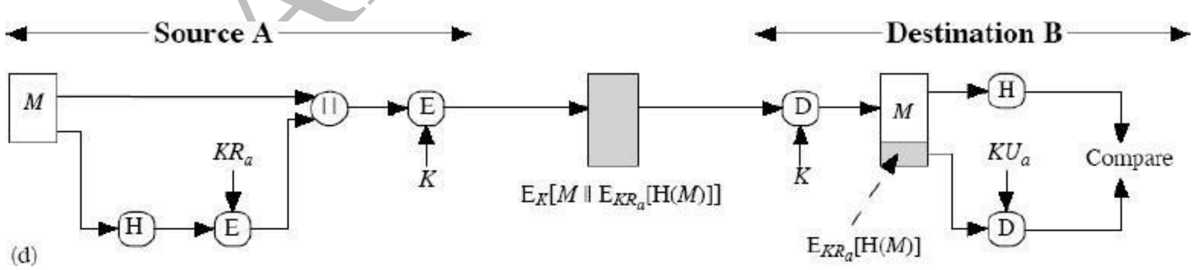


In Fig (c) Only the hash code is encrypted, using the public key encryption and using thesender"s private key.Itprovidesauthenticationplusthedigitalsignature.

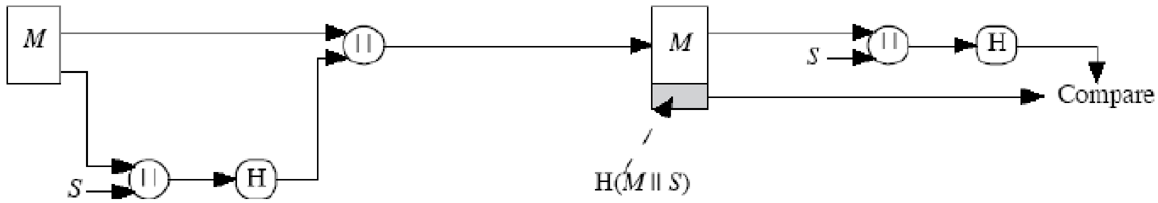


Figure(b &c)Basicuse ofHashFunction

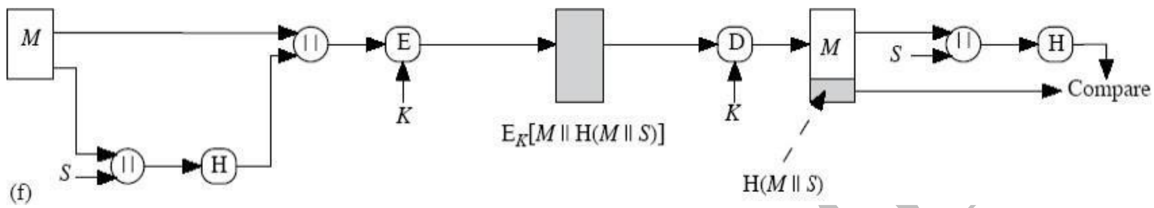
In Fig(d) If confidentiality as well as digital signature is desired, then the message plus thepublickey encryptedhashcodecanbeencryptedusinga symmetricsecretkey.



In Fig (e) This technique uses a hash function, but no encryption for message authentication.This technique assumes that the two communicating parties share a common secret value „S".The source computes the hash value over the concatenation of M and S and appends theresultinghash valuetoM.



InFig(f)Confidentialitycanbeaddedtothepreviousapproachbyencryptingtheentiremessageplusthe hash code.



Figure(d,e&f)Basicuse ofHash Function

AhashvaluehisgeneratedbyafunctionHoftheform

$$h = H(M)$$

whereMisa variable-lengthmessageand H(M)isthefixed-lengthhash value.

The hash value is appended to the message at the source at a time when the message isassumed or known to be correct. The receiver authenticates that message by recomputing thehashvalue.

RequirementsforaHashFunction

1. H canbeappliedtoablockofdataofanysize.

2. H produces a fixed-lengthoutput.

3. H(x)isrelativelyeasyto computeforanygivenx,making bothhardwareandsoftwareimplementationspractical.

4. For any given value h, it is computationally infeasible to find x such that H(x) = h. This issometimesreferred to intheliteratureas the one-wayproperty.

5. For any given block x, it is computationally infeasible to find y x such that H(y) = H(x). This issometimes referred to as weakcollisionresistance.

6. It is computationally infeasible to find any pair (x, y) such that H(x) = H(y). This is sometimesreferredto asstrongcollisionresistance. Thefirstthreepropertiesarerequirementsforthepracticalapplicationofahashfunctiontomessageauthentication.

Thefourthproperty,theone-way property,statesthatitiseasyto generateacodegivenamessagebutvirtuallyimpossibletogenerate a messagegivenacode.

Thefifthpropertyguaranteesthatanalternativemessagehashingtothesamevalueasagivenm essagecannot be found.Thispreventsforgerywhenanencryptedhashcode isused.

Thesixthpropertyreferstohowresistantthehashfunctionistoatypeofattackknownasthe birthday attack.

SimpleHashFunctions

All hash functions operate using the followinggeneral principles. The input (message,file, etc.) is viewed as a sequence of n-bit blocks. The input is processed one block at a time inan iterative fashion to produce an n-bit hash function. One of the simplest hash functions is thebit-by-bitexclusive-OR(XOR)ofeveryblock.

This can be expressed as follows: $C_i = b_{i1} + b_{i1} \ldots + b_{im}$ where

C_i=i^{th}bitof thehashcode,1≤i≤n
m =numberofn-bitblocksin
theinputb$_{ij}$=ith bitinjthblock

Procedure:

1. Initiallysetthen-bit hashvaluetozero.

2. Processeachsuccessiven-bitblockofdataasfollows:

   a. Rotate the currenthash valuetothe leftbyonebit.
   b. XOR theblockintothehashvalue.

BirthdayAttacks

Suppose that a 64-bit hash code is used. One might think that this is quite secure. Forexample,ifanencryptedhashcodeCistransmittedwiththecorrespondingunencryptedmessage M, then an opponent would need to find an M' such that H(M') = H(M) to substituteanothermessageandfoolthe receiver.

Onaverage,theopponentwouldhavetotryabout2^{63}messagestofindonethatmatches thehash codeoftheintercepted message.

However, a different sort ofattack is possible, based on the birthday paradox. Thesource, A, is prepared to "sign" a message by appending the appropriate m-bit hash code andencryptingthathash codewithA'sprivatekey.

1. The opponent generates $2^{m/2}$ variations on the message, all of which convey essentially thesame meaning.(Fraudulentmessage)

2. The two sets of messages are compared to find a pair of messages that produces the samehashcode.Theprobabilityofsuccess,bythebirthdayparadox,isgreaterthan0.5.Ifnomatchisfound,additionalvalidandfraudulentmessagesaregenerateduntilamatchismade.

3. The opponent offers the valid variation to A for signature. This signature can then be attachedto the fraudulent variation for transmission to the intended recipient. Because the two variationshave the same hash code, they will produce the same signature; the opponent is assured ofsuccesseven thoughthe encryptionkey isnotknown.

Thus,ifa64-bit hashcodeisused, thelevel ofeffort requiredisonlyonthe orderof 2^{32}

MEET-IN-THE-MIDDLEATTACK.

Divide a message M into fixed-size blocks $M_1, M_2, \ldots, M_N$ and use a symmetric encryption systemsuchas DESto compute the hashcodeGasfollows:

$$H_o = \text{initial value}$$
$$H_i = E_{M_i}[H_{i-1}]$$
$$G = H_N$$

This is similar to the CBC technique, but in this case there is no secret key. As with any hashcode, this scheme is subject to the birthday attack, and if the encryption algorithm is DES andonly a64-bithashcode isproduced,thenthe systemisvulnerable.

Furthermore, another version ofthe birthday attack can be used even ifthe opponenthasaccess toonly one messageanditsvalid signatureandcannotobtain multiplesignings.

Here is the scenario; we assume that the opponent intercepts a message with a signature in theformofanencryptedhashcodeandthatthe unencryptedhash codeis mbitslong:

1. Calculatetheunencrypted hash codeG.

2. Constructanydesired message intheform $Q_1, Q_2, \ldots, Q_{N2}$.

3. Computefor $H_i = EQ_i\ [H_{i-1}]$ for $1 \le i \le (N-2)$.

4. Generate $2^{m/2}$ random blocks; for each block X, compute $E_X[H_{N-2}.]$ Generate an additional $2^{m/2}$ randomblocks;foreachblockY,compute $D_Y[G]$,whereDisthedecryptionfunctioncorrespondingto E.

5. Basedonthebirthdayparadox,withhighprobabilitytherewillbeanXandYsuchthat E_X $[H_{N-2}] = D_Y[G]$.

6. Form the message $Q_1, Q_2, \ldots, Q_{N-2}$,X,Y. Thismessage hasthe hash code Gand thereforecanbeused withthe intercepted encryptedsignature.

4.4. SECURITYOFHASHFUNCTIONANDMAC

Just as with symmetric and public-key encryption, we can group attacks on hash functions andMACsintotwocategories:brute-forceattacksandcryptanalysis.

Brute-ForceAttacks

Thenatureofbrute-forceattacksdifferssomewhatfor hashfunctionsandMACs.

HashFunctions

Thestrengthofahashfunction againstbrute-force attacks depends solelyonthelengthofthehashcodeproduced by thealgorithm.

RequirementsofHashFunction:

One-way:Foranygivencode h, it iscomputationallyinfeasibletofindxsuchthatH(x)= h.

Weak collision resistance: For any given block x, it is computationally infeasible to findy xwithH(y)=H(x).

Strong collision resistance: It is computationally infeasible to find any pair (x, y) suchthatH(x)=H(y).

Forahash codeoflengthn,thelevelofeffort required,aswehaveseen isproportionalto thefollowing:

Oneway	2^n
Weakcollisionresistance	2^n
Strongcollisionresistance	$2^{n/2}$

MessageAuthenticationCodes

A brute-force attack on a MAC is a more difficult undertaking because it requires knownmessage-MAC pairs..To attacka hash code,wecan proceed in the followingway.Given afixed message x with n-bit hash code h = H(x), a brute-force method of finding a collision is topickarandombitstringyandcheckifH(y)=H(x). Theattackercando thisrepeatedlyoffline.

To proceed, we need to state the desired security property of a MAC algorithm, whichcanbeexpressed asfollows:

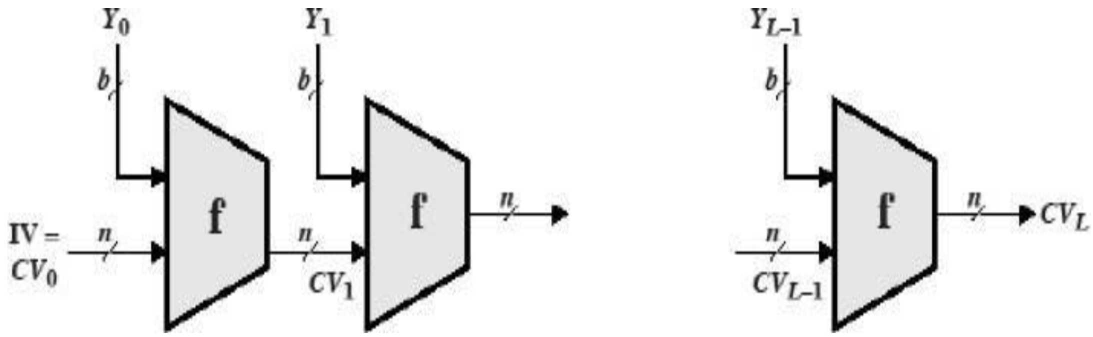
Cryptanalysis

Aswithencryptionalgorithms,cryptanalyticattacksonhashfunctionsandMACalgorithms seek to exploit some property of the algorithm to perform some attack other than anexhaustivesearch.

HashFunctions

The hash function takes an input message and partitions it into L fixed-sized blocks of bbits each. If necessary, the final block is padded to b bits. The final block also includes the valueofthetotallengthoftheinputtothehashfunction(Fig3.4).Theinclusionofthelengthmakesthejobof theopponent moredifficult.

Either theopponent mustfindtwo messagesofequal lengththathashto thesamevalueor twomessagesofdifferinglengthsthat, together with theirlengthvalues, hashto thesamevalue.



IV=InitialValue
Y_i=ithinputblock
n=LengthofHashcode

CV=ChangingVariable
L=numberofinputblocks
b=Lengthofinputblock

General structure of secure hashcode

Thehashalgorithminvolvesrepeateduseofacompression function,f,thattakestwoinputs (an n-bit input from the previous step, called the chaining variable, and a b-bit block) andproducesan n-bitoutput.

At the start of hashing, the chaining variable has an initial value that is specified as part ofthe algorithm. The final value of the chaining variable is the hash value. Often b > n; hence thetermcompression.

Thehash functioncanbesummarizedasfollows:

CV_0 = IV = initial n-bit
valueCV_I = f(CV_{i-1}, Yi_{-1}) 1 ≤i
≤LH(M)=CV_L
Where the input to the hash function is a message M consisting of the blocks Y_o, Y_1,..., Y_{L-1}. Thestructurecan be usedtoproduce asecurehashfunctiontooperate on amessageofanylength.

Message Authentication Codes :

There is much more variety in the structure of MACs than in hash functions, so it is difficultto generalizeaboutthe cryptanalysisofMACs.

4.5. SHA

The algorithm takes as input a message with a maximum length of less than bits andproducesasoutput a 512-bit messagedigest.The inputisprocessedin 1024-bitblocks.Figure **3.1**depicts theoverallprocessingofa messagetoproduceadigest.

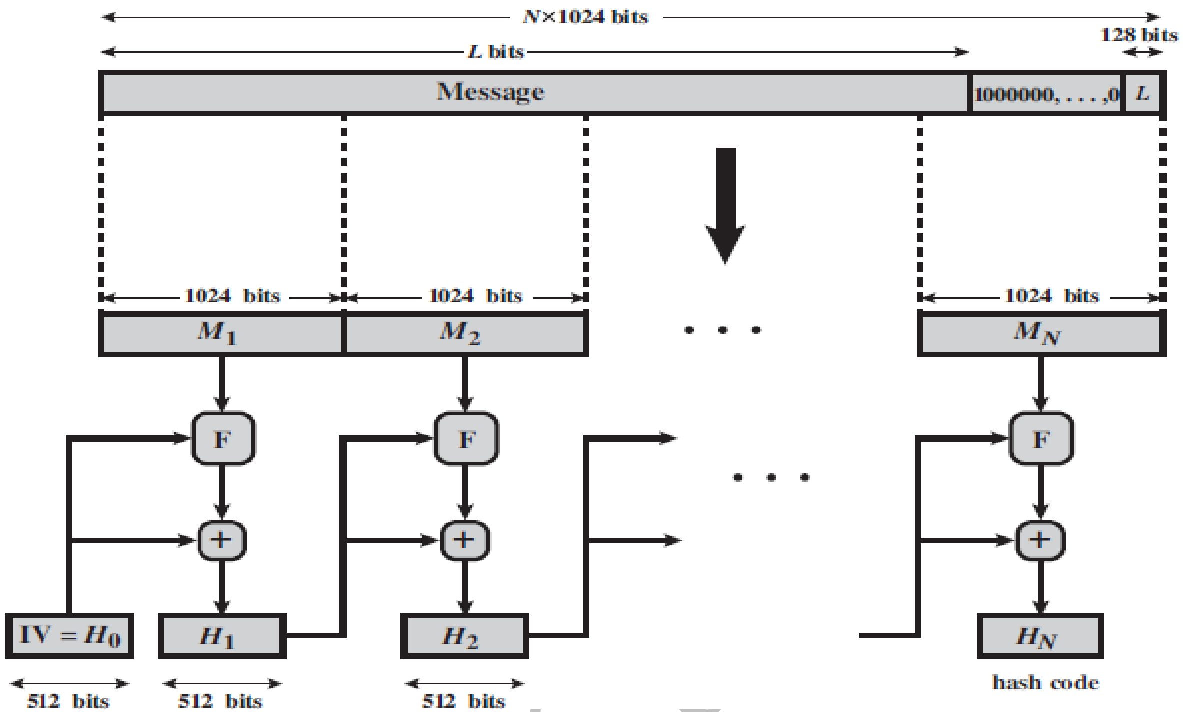


Fig.MessageDigestGenerationUsingSHA-512

The processing consists of the following

steps.Step1:Appendpadding bits.

The message is padded so that its length is congruent to 896 modulo 1024. Padding isalways added, even ifthe message is already of the desired length. Thus, the number ofpadding bits is in the range of 1 to 1024. The padding consists of a single 1 bit followed by thenecessary numberof0bits.

Step2:Appendlength.

A block of 128 bits is appended to the message. This block is treated as an unsigned128-bit integer (most significant byte first) and contains the length of the original message(before thepadding).

The outcome of the first two steps yields a message that is an integer multiple of 1024bits in length. In Figure 3,8 , the expanded message is represented as the sequence of 1024-bitblocksM_1,M_2, .. M_N,sothatthetotallengthoftheexpandedmessageisNX1024bits.

Step 3:Initialize hash buffer.

A 512-bit buffer is used to hold intermediate and final results of the hash function.Thebuffer can be represented as eight 64-bit registers (a, b, c, d, e, f, g, h).These registers areinitializedtothefollowing64-bitintegers (hexadecimalvalues):

a=6A09E667F3BCC908
e=510E527FADE682D1b
=BB67AE8584CAA73B
f=9B05688C2B3E6C1Fc=
3C6EF372FE94F82B g =
1F83D9ABFB41BD6Bd=A54FF53A5F1D36F1
h=5BE0CD19137E2179

These values are stored in big-endian format, which is the most significant byte of a word in thelow-address (leftmost) byte position. These words were obtainedby takingthe first sixty-fourbitsofthe fractionalpartsofthesquare roots ofthefirsteightprimenumbers.

Step4:Processmessagein1024-bit (128-word)blocks.

The heart of the algorithm (Fig 3.9) is a module that consists of 80 rounds; Each roundtakesas input the 512-bitbuffer value,abcdefgh,andupdatesthe contentsof thebuffer. Atinputtothefirstround,thebuffer has thevalueoftheintermediatehashvalue,H_{i-1}

Each round makes use of a 64-bit value W_t, derived from the current 1024-bit block (M_i)beingprocessed.Thesevalues arederived usingamessage scheduledescribed subsequently.

Each round also makes use of an additive constant k_t, where 0<=t<=79 indicates one ofthe80 rounds.

The output of the eightieth round is added to the input to the first round (H_i-1) to produceH_i.The addition is done independently for each of the eight words in the buffer with each of thecorrespondingwordsinHi-1,usingadditionmodulo264.

Step5:Output.

After all N 1024-bit blocks have been processed, the output from the Nth stage is the 512-bitmessagedigest.

ThebehaviorofSHA-1 is summarizedasfollows:

H_0=IV

H_i = SUM_{64} (H_{i-1},

ABCDEFGH_i)MD=H_N

Where

IV =initial value oftheabcdefghbuffer, definedinstep3

$ABCDE_q$=theoutputofthelastroundofprocessingofthe*i*th messageblock

L = the number of blocks in the message (including padding andlength fields)

SUM_{32} =Additionmodulo2^{32}performedseparatelyoneachwordofthepairof inputs

MD = finalmessagedigestvalue

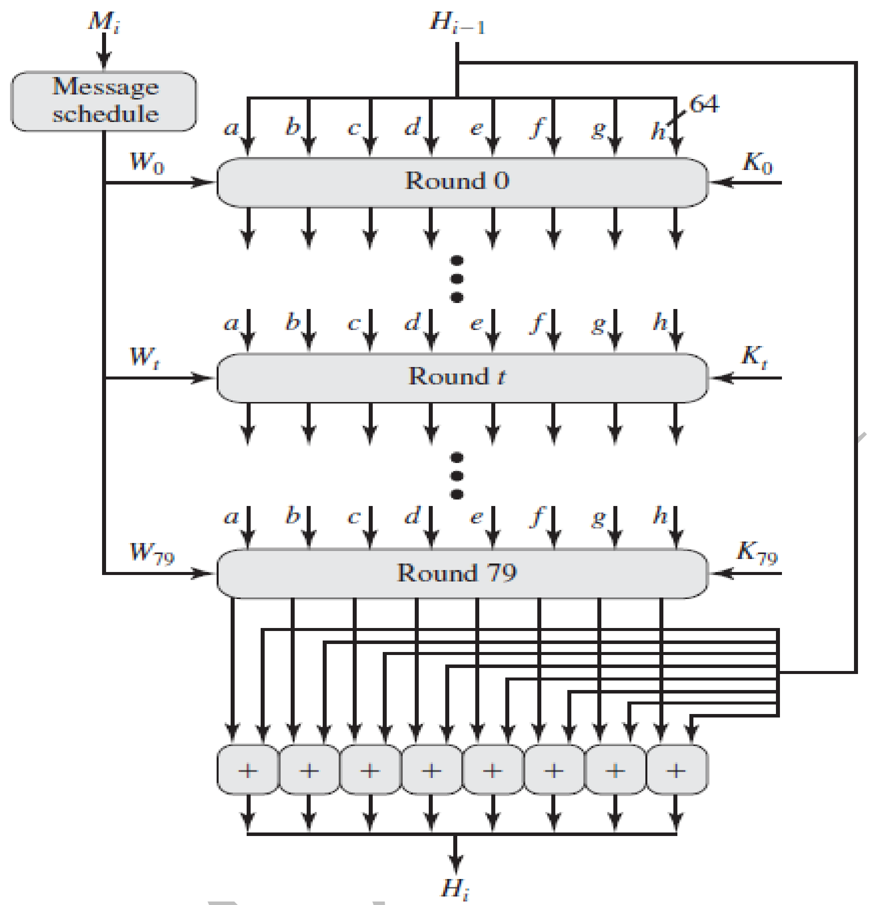


Figure. SHA-512 Processing of a Single 1024-Bit

BlockSHA-512Round Function

Letuslookin moredetailatthelogic ineachofthe 80steps oftheprocessingofone512-bitblock.Each roundisdefinedbythe followingsetofequations:

T_1=h+Ch(e,f,g)+(Σ_1^{512}e)+Wt+Kt)

T_2 =(Σ_0^{512} a)+Maj(a,b,c)

h= g g= f f=e e=d+ T_1 d=c
c=b b=a a= T_1+T_2

Where

T =Stepnumber;0≤

t≤79Ch(e,f,g)=(aAND f)$\oplus$(NOTeANDg)

Theconditionalfunction:Ife then felseg(Fig3.10)

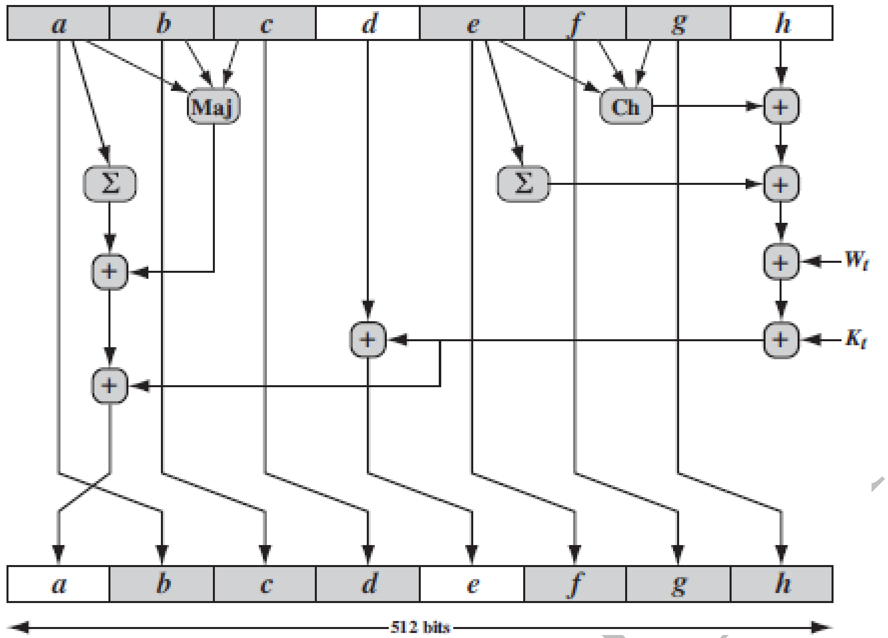


Fig.ElementarySHAOperation(singlestep)

Thefunctionscanbesummarizedasfollows:

Steps	FunctionName	FunctionValue
0≤t≤9	f1= f(t,B,C,D)	(B∧C)V (B!∧D)
20≤t≤39	f2= f(t,B,C,D)	B⊕C⊕D
40≤t≤59	f3= f(t,B,C,D)	(B∧C)V (B∧D)V (C∧ D)
60≤t≤79	f4= f(t,B,C,D)	B⊕C⊕D

ThelogicaloperatorsAND,OR,NOT,XOR,arerepresentedby the symbols∧V!⊕Only

threedifferent functionsareused.

For,0≤t≤19thefunctionistheconditionalfunction.

For20≤ t≤39and60≤t ≤79thefunctionproducesa paritybit.

ForA0≤t ≤ 59thefunctionis trueif twoor threeoftheargumentaretrue.

The following diagram illustrates how the 32bit word values wt are derived from the 512 bitmessage.

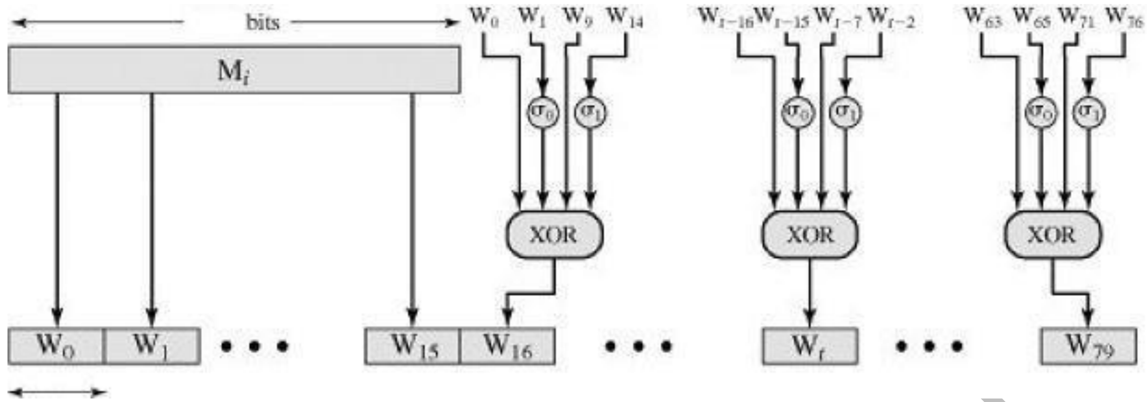


Figure.Creationof 80-wordInput SequenceforSHA-512ProcessingofSingleBlock

The first 16 values of wt are taken directly from the 16 words of the current block.the remainingvaluesaredefinedasfollows.

$$w_t=S^{1}(w_{t-16}+w_{t-14}+w_{t-8}+w_{t-3})$$

Thus in the first 16 steps ofprocessing the values ofwt is equal to the corresponding word inthemessageblock.Fortheremaining64stepsthevalueofwtconsistsofthecircularleftshiftbyonebitofthe XORoffourofthe processingvaluesofw_t.

Both MD5 and RIPEMD-160 uses one of the 16 words of a message block directly as input toeachstepfunctiononly theorderofthewordispermuted fromround toround.

SHA-1 expandsthe16blockwordsto80words foruseinthecompressionfunction.

ComparisonofSHA-1and MD5

Becausebothare derivedfrom MD4,SHA-1and MD5aresimilar tooneanother.

1. Securityagainstbrute-forceattacks:

Themostimportant difference isthat theSHA-1 digestis32bitslonger thantheMD5 digest.

Using a brute force technique the difficulty of producing any message having a given messagedigestisonthe orderof2^{128}operationsforMD5and2^{160}forSHA-1.

Using brute force technique the difficulty of producing two messages having the same messagedigest is on the order of 2^{64} operations for MD5 and 2^{80} for SHA-1.Thus SHA-1 is considerablystrongeragainstbruteforceattacks.

2. Securityagainstcryptanalysis:

MD5isvulnerabletocryptanalytic attacks.

SHA-1 isnotvulnerable tosuchattacks.

3. Speed:

Bothalgorithmsrelyonadditionmodule 2^{32}, so both do well on32bitarchitecture

SHA-1 involves more steps (80) and must process a 160 bit buffer compared to MD5‟s 128 bitbuffer.

Thus SHA-1should execute moreslowlythanMD5onthesamehardware.

4. Simplicityandcompactness:

Both algorithms are simple to describe and simple to implement and do not require largeprogramsorsubstitution tables.

5. Littleendian versusbigendianarchitecture:

MD5 usesalittleendianschemeandSHA-1uses abigendianscheme.

HMAC

HMACDesignObjectives:

- Tousehashfunctionsthatperform wellinsoftwareand for which codeisfreely andwidelyavailable.
- Toallowfor easyreplacementoftheembeddedhashfunctionin case fasterormoresecurehashfunctionsarefound orrequired.
- To preserve the original performance of the hash function without incurring a significantdegradation.
- Touseandhandlekeysinasimpleway.
- Tohave awell understoodcryptographicanalysisofthestrengthof theauthenticationmechanismbasedonreasonableassumptionsabouttheembeddedhas hfunction.

The first two objectives are important to the acceptability of

HMAC.HMACtreatsthehashfunction asa"blackbox."Thishastwo benefits.

First,anexistingimplementationofahashfunctioncanbeusedasamoduleinimplementing HMAC. In this way, the bulk of the HMAC code is prepackaged and ready to usewithoutmodification.

Second,ifitiseverdesiredtoreplaceagivenhashfunctioninanHMACimplementation,removet heexistinghashfunctionmoduleand drop in thenewmodule.

HMACAlgorithm:

Definitionoftermsusedinalgorithm(Fig3.12).

H =embeddedhashfunction(e.g., MD5,SHA-1, RIPEMD-160)

IV= initial valueinputtohashfunction

M=message inputto HMAC

Y_i=$i$thblockof$M$,0≤i≤(L–1)

L=numberofblocksin$M$

b=numberofbitsin a block

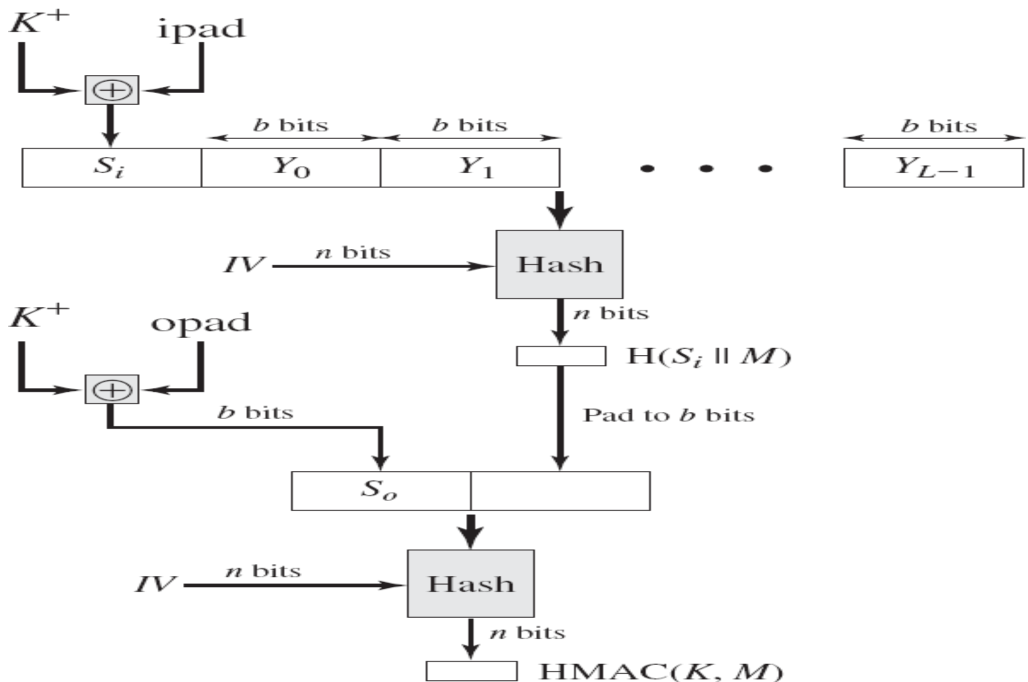


Figure.HMACStructure

n=lengthofhashcodeproducedby embedded hashfunction

K=secret key;recommendedlengthis n;ifkey lengthisgreater than$b$,thekeyisinputtothehashfunction toproducean$n$-bitkey

K^+=Kpaddedwithzerosontheleftso thattheresultis$b$bits

inlengthipad=00110110 (36inhexadecimal)repeated$b$/8times

opad=01011100(5Cin hexadecimal)repeated$b$/8 times

Then HMACcanbeexpressedas

$$\text{HMAC}(K,M) = \text{H}[(K^+ \oplus \text{opad}) \,||\, \text{H}[(K^+ \oplus \text{ipad}) || M]]$$

Wecan describethealgorithmasfollows:

1. Append zerostotheleftend oftocreate a-bitstring(e.g.,ifisoflength 160bitsand, thenwillbeappendedwith44zeroes).

2. XOR(bitwise exclusive-OR)with ipadto producetheb -bitblockS_i.

3. AppendM toS_i.

4. ApplyHtothestreamgenerated in step3.

5. XORK^+withopad toproducethe b-bitblockS_0.

6. Appendthe hashresult fromstep 4toS_0.

7. Apply Htothestreamgenerated instep6 andoutputtheresult.

Amoreefficientimplementationispossible.Twoquantitiesareprecomputed:

$$f(IV,(K^+ \oplus ipad))$$

$$f(IV,(K^+ \oplus opad))$$

In effect, the precomputed quantities substitute for the initial value (IV) in the hashfunction. With this implementation (Fig 3.13), only one additional instance of the compressionfunctionisadded totheprocessingnormallyproducedbythe hash function.

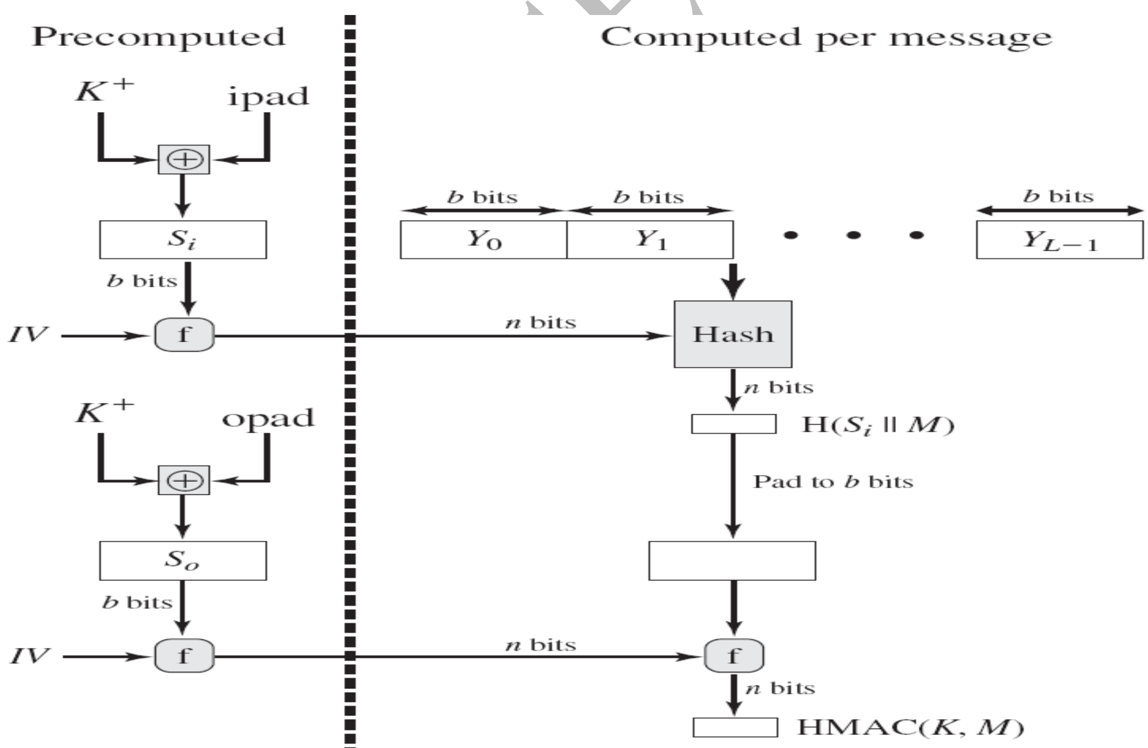


Figure.EfficientImplementationofHMAC

Securityof HMAC

The security of a MAC function is generally expressed in terms of the probability ofsuccessful forgery with a given amount of time spent by the forger and a given number ofmessage-tagpairscreatedwiththesamekey.

In essence, it is proved in that for a given level of effort (time, message-tag pairs) onmessages generated by a legitimate user and seen by the attacker, the probability of successfulattackonHMACisequivalenttooneofthefollowingattacks ontheembeddedhash function.

1. Theattacker isabletocomputeanoutputofthecompressionfunctionevenwith anthatisrandom,secret,andunknowntotheattacker.

2. Theattackerfindscollisionsinthe hashfunctionevenwhentheIVisrandomandsecret.

In the first attack, we can view the compression function as equivalent to the hashfunctionappliedtoamessageconsistingofasinglebbitblock.Forthisattack,theIVofthehash function is replaced by a secret, random value of bits. An attack on this hash functionrequires either a brute-force attack on the key, which is a level of effort on the order of 2^n, or abirthdayattack.

In the second attack, the attacker is looking for two messages M & M"and that producethesamehash:H(*M*)=H(*M*").

CMAC

Only messages of one fixed length of*mn* bits are processed, where*n* is the cipher blocksizeand *m* is a fixed positive integer. a simple example, notice that given the CBC MAC of a one-block message *X*, say *T* = MAC(*K*, *X*), the adversary immediately knows the CBC MAC for thetwo blockmessage $X \| (X \oplus T)$sincethisisonceagain*T*.

Black and Rogaway [BLAC00] demonstrated that this limitation could be overcome using threekeys: one key *K* of length *k* to be used at each step of the cipher block chaining and two keys oflength *b*,where*b* is the cipherblocklength.

The **Cipher-based Message Authentication Code** (CMAC) mode ofoperation for use withAESandtripleDES.Itis specifiedinNISTSpecialPublication800-38B.

First, let us define the operation of CMAC when the message is an integer multiple *n* of thecipherblocklength*b*.ForAES,*b*=128,andfortripleDES,*b*=64.Themessageisdividedinto*n* blocks (*M*1, *M*2,c, *Mn*). The algorithm makes use of a *k*-bit encryption key *K* and a *b*-bitconstant, *K*1. For AES, the key size *k* is 128, 192, or 256 bits; for triple DES, the key size is 112or168 bits.CMACiscalculatedasfollows

$$
\begin{aligned}
C_1 &= \mathrm{E}(K, M_1) \\
C_2 &= \mathrm{E}(K, [M_2 \oplus C_1]) \\
C_3 &= \mathrm{E}(K, [M_3 \oplus C_2]) \\
&\cdot \\
&\cdot \\
&\cdot \\
C_n &= \mathrm{E}(K, [M_n \oplus C_{n-1} \oplus K_1]) \\
T &= \mathrm{MSB}_{Tlen}(C_n)
\end{aligned}
$$

where

T=messageauthenticationcode,alsoreferredtoas thetag
Tlen=bit lengthofT
MSB*s*(*X*)= the*s*leftmostbitsof thebitstring*X*

TheCMACoperation(Fig3.14)thenproceeds as before,exceptthatadifferent*b*-bitkey*K*2isusedinstead of*K*1.

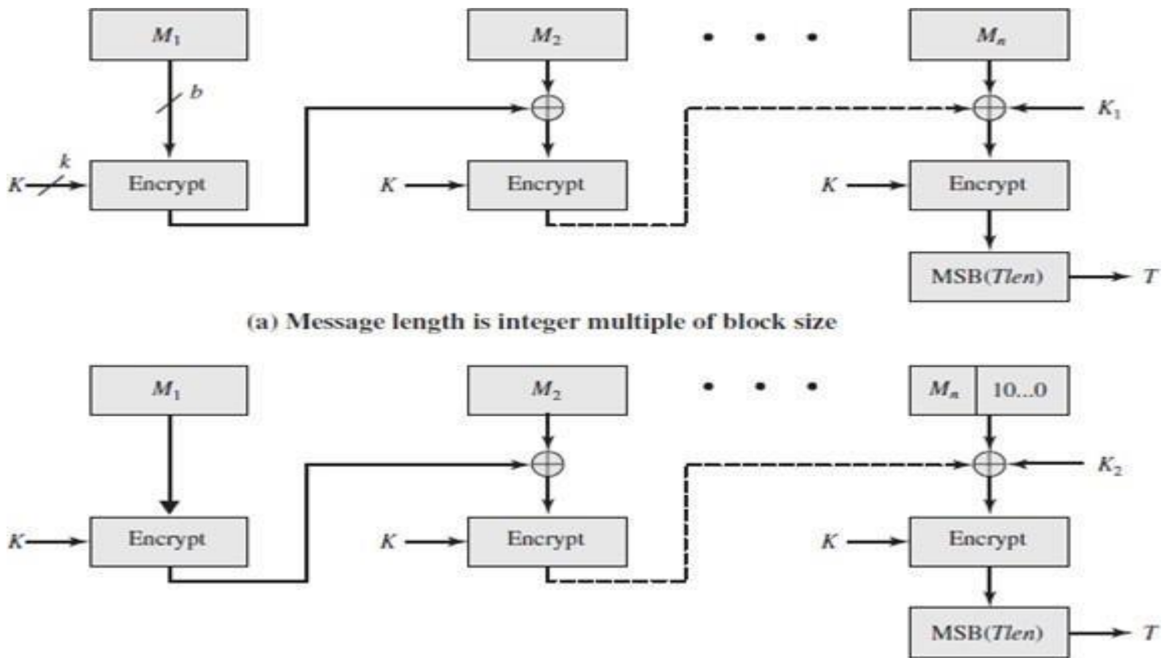


Fig.Cipher-basedMessage AuthenticationCode

Thetwo*b*-bitkeysare derivedfromthe*k*-bitencryption keyasfollows.

$$L = E(K, 0^b)$$
$$K_1 = L \cdot x$$
$$K_2 = L \cdot x^2 = (L \cdot x) \cdot x$$

wheremultiplication(#)isdonein thefinitefield GF(2*b*)and *x* and*x*2arefirstand second-order polynomialsthatare elementsof GF(2*b*).Thus, thebinaryrepresentation of*x*consists of*b*-2zeros followedby10;thebinary representationof*x*2consists of*b*-3zeros followedby100.

4.6. DIGITALSIGNATUREANDAUTHENTICATIONPROTOCOLS

DigitalSignatureRequirements

Message authentication protects two parties who exchange messages from any third party.However,itdoesnotprotectthetwoparties againsteach other.

Disputes createdbymessageauthenticationare:

- Creationof fraudmessage.

- Denythesendingof message

Forexample,supposethatJohnsendsanauthenticatedmessagetoMary,thefollowingdisputes thatcouldarise:

1MarymayforgeadifferentmessageandclaimthatitcamefromJohn.Marywouldsimplyhave to create a message and append an authentication code using the key that John and Maryshare.

2. John candenysendingthemessage.Becauseitispossible forMary to forgeamessage,thereisnoway toprovethatJohndid infactsend themessage.

<u>Propertiesof digitalsignature:</u>

- Itmustverify theauthor andthe date andtime ofthesignature.
- Itmusttoauthenticatethecontentsatthe timeofthesignature.
- Itmustbeverifiable bythirdparties, toresolvedisputes.

<u>Requirementsforadigital signature:</u>

- Thesignaturemustbeabitpatternthatdependsonthemessagebeingsigned.
- The signature must use some information unique to the sender, to prevent both forgeryanddenial.
- It mustberelatively easytoproducethedigitalsignature.
- It mustberelativelyeasytorecognize and verifythedigitalsignature.
- It must be computationally infeasible to forge a digital signature, either by constructing anew message for an existing digital signature or by constructing a fraudulent digitalsignature foragivenmessage.
- It mustbepracticaltoretain acopyofthedigital signature instorage.

DirectDigital Signature

The term **direct digital signature** refers to a digital signature scheme that involves only thecommunicating parties (source, destination). It is assumed that the destination knows the publickeyofthesource.

Confidentiality can be provided by encrypting the entire message plus signature with ashared secret key (symmetric encryption). Note that it is important to perform the signaturefunction first and then an outer confidentiality function. In case of dispute, some third party mustviewthemessage anditssignature.

If the signature is calculated on an encrypted message, then the third party also needsaccess to the decryption key to read the original message. However, if the signature is the inneroperation, then the recipient can store the plaintext message and its signature for later use indisputeresolution.

Thevalidityoftheschemejustdescribed dependsonthe securityofthe sender‟sprivatekey.

WeaknessofDirectDigitalSignature:

- If a sender later wishes to deny sending a particular message, the sender can claim thattheprivatekeywaslostorstolenandthatsomeoneelseforged hisorhersignature.

- Another threat is that some private key might actually be stolen from X at time T. Theopponent can then send a message signed with X‟s signature and stamped with a timebeforeorequaltoT.

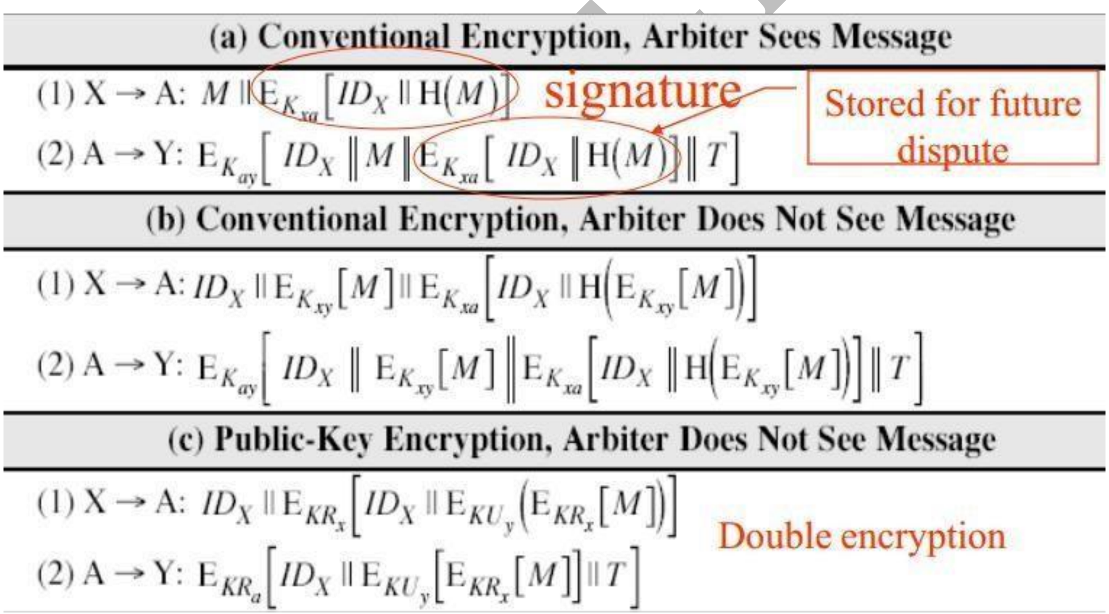
ArbitratedDigitalSignatures

The problem associated with the Direct digital signature can be overcome by using arbitratedschemes.

In the arbitrated scheme, the entire signed message from the sender goes to the arbiter A. Thearbiter subjects the message and signature to a number of tests to check the origin and control.The date and time is attached to the message. This indicates that the digital signature has beenverifiedand issatisfied. The messageisthentransmitted tothereceiver.

Requirementofthearbiter:

- Asthearbiterplaysasensitive and crucial role,it should beatrustedthirdparty.



Notation:X=SenderY=RecipientA = ArbiterM=MessageT=Timestamp

Scheme1:Conventionalencryption, Arbiterseesthemessage:

Thesender XandarbiterAshare the masterkey K_{ax} thereceivery andthearbiterAsharethe masterkey K_{ay}

When Xwants to senda message M to Y, construct a message computes the hashvalue H(M). This hash is encrypted using symmetric encryption with the key Kax which acts assignature.The messagealongwiththesignatureistransmitted toA.

AtA,itdecryptsthesignatureandchecks thehashvaluetovalidatethemessage.Atransmit themessage toY,encryptedwith K_{ay}.Y decrypttoextractthemessageand signature.Disadvantage:

Eaves droppercanread themessageasthereis noconfidentiality.

Scheme2:Conventional encryption, Arbiterdoesnotseethemessage:

- K_{ax} andK_{ay}arethe masterkeys.
- K_{xy} isthe key shared betweentheXandY
- When xwantstotransmitamessage toY,thepacketgoestoarbiter.
- Thesame procedure as thatofIschemeisusedXtransmitanidentifier,acopy ofthemessageencryptedwithK_{xy}andasignature toA.
- Thesignatureisthehashofthemessage encrypted withK_{xa}
- Adecryptthesignature,andchecks thehashvaluetovalidatethemessage.
- Acannot read themessage,Aattaches to itthetimestamps,encryptwith K_{xa}andtransmittoY.

Attack:Thearbitercanjoinwith an attackerand denya messagewithsender"ssignature.

Scheme2:Public keyencryption,Arbiter doesnotseethemessage:

This method uses the public key cryptography which gives authentication and digital signature.Thedoublyencryptedmessageisconcatenated withID_x andsent toarbiter.

- Acandecrypttheouterencryptionto ensurethatthe messagehascomefromX.
- Athen transmitthe

messagewithID_xandtimestamp.Advantages:

- Noinformation issharedamongpartiesbefore communication,hencefraudisavoided.
- Noincorrectly dated

messagecanbesent.Disadvantages:

Thecomplexpublickeyalgorithmis tobetwiceforencryption and twice fordecryption.

AuthenticationProtocols

Authentication Protocols used to convince parties of each other"s identity and to exchangesession keys.

MutualAuthentication

An important application area is that of mutual authentication protocols. Such protocolsenable communicating parties to satisfy themselves mutually about each other"s identity and toexchangesessionkeys.

Keyissuesare

- **confidentiality**– toprotectsessionkeysandpreventmasqueradedandcompromised
- **timeliness**-topreventreplayattacks

ReplayAttacks
Whereavalid signed messageiscopiedand laterresent

- Simplereplay
  Theopponentsimplycopiesthemessageandreplaysitlater.

- Repetition thatcanbelogged
  Theopponent replayatime stampedmessagewithinavalid timewindow.

- Repetitionthatcannotbedetected
  Theattacker wouldhavesuppressed the original message from the receiver.Only the replay message alonearrives.

- Backwardreplaywithoutmodification
  This replay back to the sender itself.This is possible if the sender cannot easilyrecognize the difference between the message sent and the message receivedbasedon the content.

Countermeasuresinclude
One approach to coping with replay attacks is to attach a sequence number to eachmessage used in an authentication exchange. A new message is accepted only if its sequencenumberisintheproperorder.

The difficulty with this approach is that it requires each party to keep track of the lastsequence number for each claimant it has dealt with. Because of this overhead, sequencenumbersare generallynotused forauthenticationand key exchange.Instead,oneofthefollowingtwogeneralapproachesisused:

- **Timestamps:** Party A accepts a message as fresh only if the message contains a**Timestamp** that is close enough to A‟s knowledge of current time. This approachrequires thatclocksamongthe variousparticipantsbesynchronized.
- **Challenge/response:** Party A, expecting a fresh message from B, first sends B a**nonce** (challenge) and requires that the subsequent message (response) receivedfromB containthe correctnoncevalue.

UsingSymmetricEncryption
- Use atwo-level hierarchy ofkeys
- Usuallywithatrusted KeyDistribution Center(KDC)
  - Eachparty sharesownmasterkeywithKDC
  - KDC generates sessionkeysusedforconnectionsbetweenparties
  - Masterkeysusedtodistributethe sessionke

Needham-SchroederProtocol

- Originalthird-partykeydistributionprotocol
- ForsessionbetweenAandB mediatedbyKDC
- Protocoloverviewis:

1. A→KDC:ID_A||ID_B||N_1
2. KDC→A:EK_a[K_s||ID_B||N_1|| EK_b[K_s||ID_A]]
3. A→B:EK_b[K_s||ID_A]
4. B→A:EK_s[N_2]
5. A→B:EK_s[f(N_2)]

Step1:AtoKDC,transmittheidofsourceanddestinationandanonceN_1asarequest.Step2:A securely acquiresthe sessionkey instep2andapackettoBencryptedwithE$K_b$isalsoreceived fromKDC.
Step3:AtransmittoBthemessageitgotfromKDC.
Step4:As ahand shake,Bencryptsanew nonceN_2andtransmitto AwithK_s.Step5:As ahand shake,AencryptthefunctionofN_2withK_s
Step4andStep 5 asusedashandshake andpreventthe replyattacks.

Attacks:

- Usedto securely distributea newsessionkeyforcommunicationsbetweenA&B
  - Butisvulnerabletoareplayattackifanoldsessionkey hasbeenCompromised
    - Then message3canberesentconvincingB thatiscommunicatingWithA
- Modificationstoaddressthisrequire:
  - Timestamps
  - Usinganextranonce

DenningProtocol:
Toovercomethe aboveweaknessbya modificationtothe Needham/Schroederprotocol thatincludestheaddition ofatimestamptosteps2and3.

A→KDC: ID_A||ID_B

KDC → A: E(K_a, [K_sIIID_B *II T II* E(K_b, [K_s *II*ID_A *II*

T])])A→B:E(K_b,[K_sIIID_A*II**T*])

B→A:E(K_s,N1)A→

B:E(K_s,f(N1))

T is a timestamp that assures A and B that the session key has only just been generated. Thus,both A and B know that the key distribution is a fresh exchange.A and B can verify timeliness bycheckingthat

$$|Clock-T|<\Delta t1+\Delta t2$$

The **Denning protocol** seems to provide an increased degree of security compared tothe Needham/Schroeder protocol. However, a new concern is raised: namely, that this newschemerequiresrelianceonclocksthatare synchronizedthroughoutthenetwork.

suppress-replayattacks:

The problem occurs when a sender‟s clock is ahead of the intended recipient‟s clock. Inthis case, an opponent can intercept a message from the sender and replay it later when thetimestamp in the message becomes current at the recipient‟s site. This replay could causeunexpectedresults.

Methodtoovercome:

One way to counter suppress-replay attacks is to enforce the requirement that partiesregularly check their clocks against the KDC‟s clock. The other alternative, which avoids theneedforclocksynchronization,istorely onhandshakingprotocolsusingnonce.

This latter alternative is not vulnerable to a suppress-replay attack, because the noncetherecipientwillchooseinthefuture areunpredictabletothe sender.

Anattemptismadeto respondto theconcernsaboutsuppress replayattacksandatthesametime fixtheproblemsintheNeedham/Schroederprotocol.

Theprotocolis

1. A: →B:ID_A||N_a
2. B: →KDC:ID_B||N_b||E(K_b,[ID_A||N_a||T_b])
3. KDC:→A:E(K_a,[ID_BƒƒN_aƒƒKsƒƒT_b])„E(Kb,[ID_A„Ks„T_b])„N_b
4. A:→B:Kb,[ID_A„K_s„T_b])„E(K_s,N_b)

1. A initiates the authentication exchange by generating a nonce,N_a , and sending that plus itsidentifier to B in plaintext. This nonce N_a will be returned to A in an encrypted message thatincludesthesession key,assuringAofitstimeliness.

2. B alerts the KDC that a session key is needed. Its message to the KDC includes its identifierand a nonce, . This nonce will be returned to B in an encrypted message that includes thesessionkey,assuringBofitstimeliness.B‟smessagetotheKDCalsoincludesablockencryptedwitht hesecretkeysharedbyBandtheKDC.ThisblockisusedtoinstructtheKDCtoissuecredentialstoA;thebl ockspecifiestheintendedrecipientofthecredentials,asuggestedexpiration timeforthecredentials,and thenonce received fromA.

3. The KDC passes on to A B‟s nonce and a block encrypted with the secret key that B shareswiththeKDC.Theblockservesasa"ticket"thatcanbeusedbyAforsubsequentauthentications, as will be seen. The KDC also sends to A block encrypted with the secret keyshared by A and the KDC. This block verifies that B has received A‟s initial message (ID_B) andthatthisisatimelymessageandnotareplay(N_a),anditprovidesAwithasessionkey(K_S)andthe time limitonitsuse(T_b).

4. A transmits the ticket to B, together with the B‟s nonce, the latter encrypted with the sessionkey.TheticketprovidesBwiththesecretkeythatisusedtodecryptE($K_{S,}$$N_b$)torecoverthe

nonce. The fact that B‟s nonce is encrypted with the session key authenticates that themessagecamefromAandisnotareplay.

UsingPublic-KeyEncryption

- Have arangeofapproachesbasedon theuseofpublic-keyencryption
- Need toensure havecorrectpublic keys forotherparties
- Usingacentral authentication server(AS)
- Variousprotocolsexistusingtimestampsornonces

DenningASProtocol

- Denningpresentedthefollowing:

    1. $A \rightarrow AS: ID_A || ID_B$
    2. $AS \rightarrow A: E_{KRas}[ID_A || KU_a || T] || EKRas[ID_B || KU_b || T]$
    3. $A \rightarrow B: E_{KRas}[ID_A || KU_a || T] || EKRas[ID_B || KU_b || T] || EKU_b[E_{KRas}[Ks || T]]$

- Notesessionkey ischosenby A,hence ASneednotbe trusted to protectit
- timestamps prevent replay but require synchronized

clocksAnotherapproachproposedbywooandlammakesuseofnonce

.

1. $A \rightarrow KDC: ID_A || ID_B$
2. $KDC \rightarrow A: E_{KRauth}[ID_B II K_{Ub}]$
3. $a \rightarrow b: EK_{Ub}[N_a II ID_A]$
4. $B \rightarrow KDC: ID_B II ID_A II EK_{Uauth}[N_a]$
5. $KDC \rightarrow B: E_{KRauth}[IDAIIKUa] II E_{KUb}[E_{KRauth}[N_a II K_s II ID_B]]$
6. $B \rightarrow A: E_{KUa}[E_{KRauth}[N_a II K_s II ID_B] II N_b]$
7. $A \rightarrow B: E_{Ks}[N_b]$

Step 1: A informs the KDC of its intention to establish a secure connection with

B.Step2:TheKDCreturns to Aa copy ofB‟s publickeycertificate.

Step3:AinformsB ofitsdesire tocommunicateandsendsa nonce N_a.

Step 4: B asks the KDC for A‟s public key certificate and request a session key.;B includes A‟snonce so that the KDC can stamp the session key with that nonce.The nonce is protected usingtheKDC‟spublickey.

Step5:TheKDC returns toBacopy ofA‟s publickeycertificate,plustheinformation$[N_a, K_s, ID_B]$.

Step 6: The triple $[N_a, K_s, ID_B]$, still encrypted with the KDC‟s private key,is relayed to A,togetherwithanonce N_b generatedby B.

AlltheforegoingareencryptedusingA‟s publickey.A retrievesthesessionkey$K_s$and uses ittoencrypt$N_b$andreturnittoB.

Step7:Assures B ofA‟sknowledgeofthesession key.

One-WayAuthentication

- Requiredwhensender&receiverarenotincommunicationsatsametime(eg.Email)
- Have headerinclearso canbedeliveredbyemailsystem
- Maywantcontentsofbodyprotected &senderauthenticated

UsingSymmetricEncryption

- canrefineuseofKDCbutcan‟thavefinalexchangeofnonce

1. $A \rightarrow KDC: ID_A || ID_B || N_1$
2. $KDC \rightarrow A: EK_a[K_s || ID_B || N_1 || EK_b[K_s || ID_A]]$
3. $A \rightarrow B: EK_b[K_s || ID_A] || EKs[M]$

- Doesnot protect againstreplays
  - couldrelyontimestampin message,thoughemaildelays makethisproblematic

Public-KeyApproaches

- Ifconfidentialityismajorconcern, canuse:

$$A \rightarrow B: EKU_b[K_s] || EK_s[M]$$

In this case, the message is encrypted with a one-time secret key. A also encrypts thisone-time key with B‟s public key. Only B will be able to use the corresponding private key torecover the one-time key and then use that key to decrypt the message. This scheme is moreefficientthansimply encryptingtheentire messagewithB‟spublic key.

- If authentication needed use a digital signature with a digital

certificate: $A \rightarrow B: M„ IIE_{KRa}(H(M))$

This method guarantees that A cannot later deny having sent the message. However, thistechnique is open to another kind of fraud. Bob composes a message to his boss Alice thatcontains an idea that will save the company money. He appends his digital signature and sendsitintothe e-mailsystem.

Eventually, the message will get delivered to Alice‟s mailbox. But suppose that Max hasheardofBob‟sideaandgainsaccesstothe mailqueuebeforedelivery.HefindsBob‟smessage, strips off his signature, appends his, and requeues the message to be delivered toAlice.MaxgetscreditforBob‟s idea.

Tocountersucha scheme,boththemessageandsignature canbeencryptedwiththerecipient‟s publickey:

$$A \rightarrow B: E_{KUb},[M|| E_{KRa},H(M)]$$

The latter two schemes require that B know A"s public key and be convinced that it is timely. Aneffectivewaytoprovidethisassurance isthedigitalcertificate

$$A \rightarrow B: M \| EKR_a[H(M)] \| EKR_{as} IIT \| ID_A \| KU_a]$$

In addition to the message,A sends B the signature encrypted with A"s private key and A"scertificateencryptedwiththeprivatekey oftheauthenticationserver.The recipientofthemessage first uses the certificate to obtain the sender"s public key and verify that it is authenticandthen usesthe publickey to verifythemessageitself.

4.7. DSS

TheDSSApproach

The DSS uses an algorithm that is designed to provide only the digital signature function. UnlikeRSA,itcannotbeusedforencryptionorkeyexchange.Nevertheless,itisapublic-keytechnique.

RSAapproach

The message to be signed is input to a hash function that produces a secure hash codeof fixed length. This hash code is then encrypted using the sender"s private key to form thesignature.

Boththemessageandthesignaturearethentransmitted.Therecipienttakesthemessageandproducesahashcode.Therecipientalsodecryptsthesignatureusingthesender"s publickey.

If the calculated hash code matches the decrypted signature, the signature is acceptedasvalid.Becauseonlythesenderknowstheprivatekey,onlythesendercouldhaveproduceda validsignature.

DSSapproach

The DSS approach also makes use of a hash function. The hash code is provided asinputtoasignaturefunctionalongwitharandom numbergeneratedforthisparticularsignature.

The signature function also depends on the sender"s private key (*PRa*) and a set ofparametersknowntoagroupofcommunicatingprincipals.Wecanconsiderthissettoconstitute a global public key (PU_G). The result is a signature consisting of two components,labeled*s*and*r(fig3.15)*.

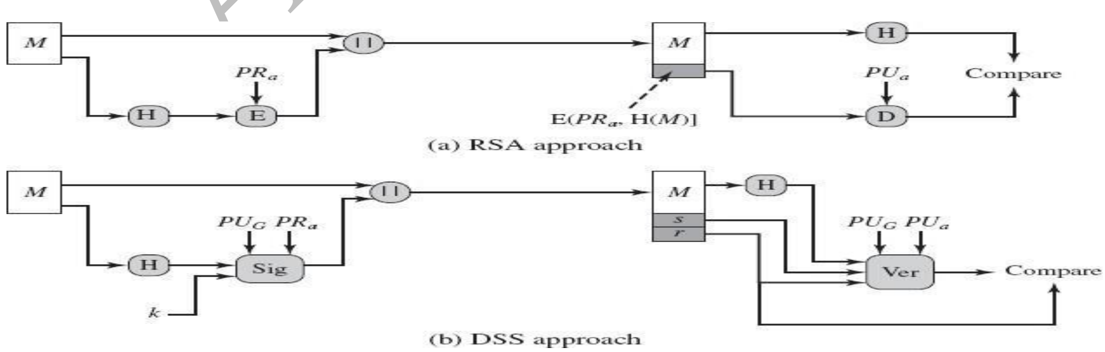


Figure3.15TwoApproachestoDigitalSignatures

At the receiving end, the hash code of the incoming message is generated. This plus thesignature is input to a verification function. The verification function also depends on the globalpublic keyaswell as thesender"spublickey,which is pairedwiththesender"sprivatekey.

The output of the verification function is a value that is equal to the signature componentif the signature is valid. The signature function is such that only the sender, with knowledge oftheprivatekey,couldhaveproducedthevalidsignature.

TheDigitalSignatureAlgorithm:

Thereare three parametersthatare publicand canbecommontoagroupofusers.

- A160-bitprime numberqischosen.
- Next, a prime number p is selectedwith alength between 512and 1024 bits suchthatqdivides(*p*-1).
- Finally,*g* is chosen to be of the form$h^{(p-1)/q}$mod *p*, where *h* is an integer between1and(p-1).

With these numbers in hand, each user selects a private key and generates a publickey.Theprivatekeyxmustbe anumber from1to (q-1) andshouldbechosenrandomly.T

The public key is calculated from the private key as y = g^x mod p .The calculation ofgiven(Fig 3.16) is relatively straightforward. However, given the public key y, it is believed to becomputationally infeasible to determine x, which is the discrete logarithm of *y* to the base *g*, modp.

Atthereceivingend,verificationisperformedusingtheformulas.Thereceivergenerates a quantity v that is a function of the public key components, the sender"s public key,and the hash code of the incoming message. If this quantity matches the component of thesignature,thenthesignatureis validated.

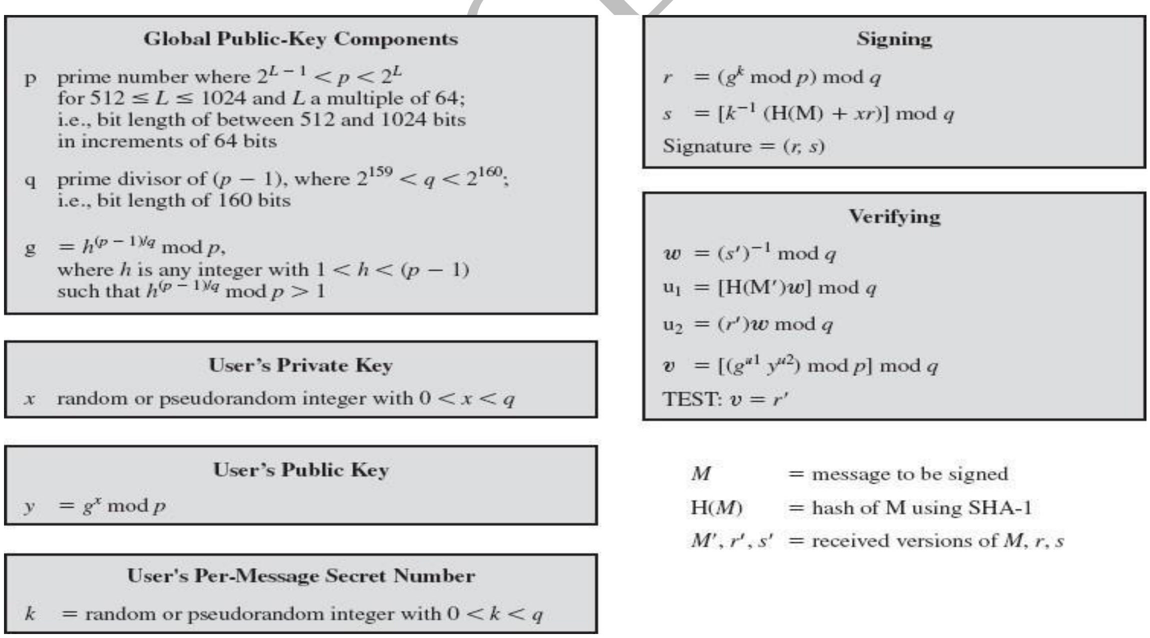


Figure.TheDigitalSignatureAlgorithm(DSA)

The value r does not depend on the message at all. Instead, r is a function of k and thethreeglobalpublic-key components.

Themultiplicativeinverseof k(modq) ispassedtoafunctionthatalsohasasinputs themessagehashcodeand theuser‟sprivatekey.

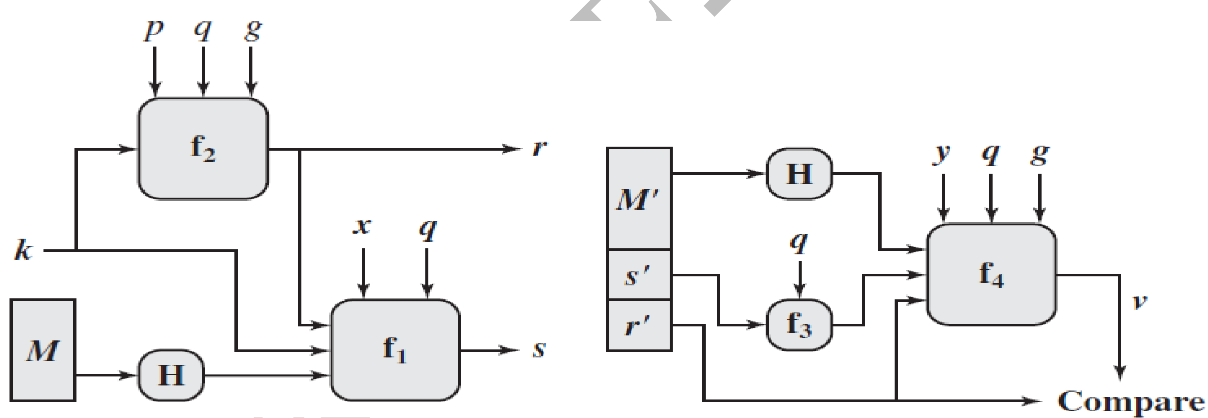
The structure of this function is such that the receiver can recover using the incomingmessage and signature, the public key of the user, and the global public key. Given the difficultyof taking discrete logarithms, it is infeasible for an opponent to recover k from r to recover x froms.

Theonlycomputationallydemandingtaskinsignaturegenerationistheexponentialcalculation g^kmod p. Because this value does not depend on the message to be signed, it can becomputedahead of time.

Selects a private key and generates a public key. The private key *x* must be a numberfrom 1 to (*q*1) and should be chosen randomly. The public key is calculated from the private keyas$y$=g^xmod$p$.

To create a signature, a user calculates two quantities, *r* and *s*, that are functions of thepublic key components (*p, q, g*), the user's private key (*x*), the hash code of the message, H(*M*),and anadditional integer*k*that shouldbegeneratedrandomly andbeunique foreachsigning.

At the receiving end, verification is performed using the formulas. Thereceiver generatesa quantity *v* that is a function of the public key components, the sender's public key, and the hashcodeoftheincomingmessage.Ifthisquantitymatchesthe*r*componentofthesignature,thenthesignatureis validated(Fig).



FigureDSSSigningandVerifying

4.8. AUTHENTICATIONAPPLICATIONS

One of the key aspects of cryptography and network security is authentication. It helps toestablish trust by identifying a particular user or a system. There are many ways to authenticateauser.Traditionally,useridsand passwordshavebeenused.

1.AuthenticationRequirements

During communicationacrossnetworks,followingattackscan beidentified.

1. **Disclosure:**Releasesofmessagecontents toanypersonorprocessnotpossessingtheappropriatecryptographickey.

2. **Trafficanalysis:**Discoveryofthe patternoftraffic betweenparties.

3. **Masquerade:**Insertionof messagesintothenetwork fraudulentsource.

4. **Content modification:** Changes to the content of the message, includinginsertiondeletion,transposition andmodification.

5. **Sequencemodification:** Any modification to a sequence of messages betweenparties,includinginsertion,deletionand reordering.

6. **Timingmodification:**Delayorreplayofmessages.

7. **Sourcerepudiation:**Denial oftransmissionof messagebysource.

8. **Destinationrepudiation:** Denialoftransmissionofmessagebydestination.

Thesecuritymeasuresfortheabovementioned attacksareasfollowsFor1,2- MessageConfidentiality

3,4,5,6 - Message Authentication

7 - DigitalSignatures

8 - Digitalsignaturewithprotocoldesignedtocountertheattack

2.AuthenticationFunctions

Any message authentication or digital signature mechanism can be viewed as havingfundamentally twolevels.

1. **Lowerlevel:**Somefunctionthatproducesanauthenticator:a valuetobeusedtoauthenticateamessage.

2. **HigherLevel:** Lowerlayerfunctions areused to createaprotocolthatenablesareceivertoverifytheauthenticityofmessage

The different types of functions that may be used to produce an authenticator are asfollows:

1. **Message encryption:** The cipher text of the entire message serves as itsauthenticator.

2. **Message AuthenticationCode(MAC):**Apublicfunctionofthemessageandasecretkey thatproducesa fixedlength valueservesas theauthenticator.

3. **Hashfunction:** Apublic functionthatmapsa messageofanylengthintoa fixedlengthhashvalue,whichservesastheauthenticator.

4.9. KERBEROS

Kerberos provides a centralized authentication server whose function is to authenticateusers to servers and servers to users. Kerberos relies exclusively on conventional encryption,makingnouseofpublic-key encryption.

Motivation

A distributedarchitecture consists ofdedicated userworkstations (clients)anddistributed orcentralizedservers.Inthisenvironment,there arethreeapproachestosecurity:

- Relyon each individual clientworkstationto assure theidentity ofitsuser orusersandrelyoneachservertoenforce asecurity policy basedonuseridentification(ID).
- Require that client systems authenticate themselves to servers, but trust the clientsystemconcerningtheidentityofitsuser.
- Require the user to prove his or her identity for each service invoked. Also require thatserversprovetheiridentityto clients.

Thefollowingare the**requirementsforKerberos**:

- **Secure:**Anetworkeavesdroppershouldnotbeabletoobtainthenecessaryinformationtoimpersonateauser.Moregenerally,Kerberosshouldbestrongenoughthatapotentialopponentdoes notfind ittobe theweaklink.
- **Reliable:** For all services that rely on Kerberos for access control, lack of availabilityof the Kerberos service means lack of availability of the supported services. Hence,Kerberos should be highly reliable and should employ distributed server architecture,withonesystemabletobackupanother.
- **Transparent:** Ideally, the user should not be aware that authentication is takingplace,beyondthe requirementto enterapassword.
- **Scalable:** The system should be capable of supporting large numbers of clients andservers.Thissuggestsamodular,distributedarchitecture.

Tosupporttheserequirements, theoverallscheme ofKerberosisthatofatrusted third-party authentication servicethatusesa protocolbasedonNeedhamandSchroeder.

It is trusted in the sense that clients and servers trust Kerberos to mediate their mutualauthentication. Assuming the Kerberos protocol is well designed, and then the authenticationserviceissecure ifthe Kerberos serveritselfissecure.

Two versions of Kerberos are in common use. **Version 4** and **Version**

5<u>**Kerberos Version 4**</u>

Version 4 of Kerberos makes use of DES, in a rather elaborate protocol, to provide theauthenticationservice

1.A SimpleAuthenticationDialogue

In an unprotected network environment, any client can apply to any server for service.The obvious security riskis that ofimpersonation. To counter this threat, servers must be abletoconfirmtheidentitiesofclientswhorequestservice.Butinanopenenvironment,thisplacesasubstantialburden on eachserver.

An alternative is to use an authentication server (AS) that knows the passwords of allusers and stores these in a centralized database. In addition, the AS shares a unique secret keywitheachserver.Thesimpleauthentication dialogue isasfollows:

1. C >> AS:$ID_c||P_c||ID_v$
2. AS>>C:Ticket
3. C>>V:ID_c||TicketTicket=E $K_v(ID_c||ADc||ID_v)$

C :Client,
AS :AuthenticationServer,
V : Server, ID_c : ID of the client,P_c :Passwordoftheclient,
AD_c : Address of client, ID_v : ID of the server,K_v :secretkey sharedbyASandV,
|| : concatenation.

2.A MoreSecureAuthenticationDialogue

Therearetwomajorproblemsassociated withtheprevious approach:

- Plaintext transmission ofthe password.
- Eachtime auserhastoenterthepassword.

Tosolve these problems,weintroduceaschemeforavoidingplaintextpasswords,andanewserver,knownasticketgrantingserver(TGS).Thehypotheticalscenarioisasfollows:

Onceper userlogon session:-

1. C>> AS:$ID_c||ID_{tgs}$
2. AS>>C:$Ek_c(Ticket_{tgs})$

Oncepertypeofservice:

3. C>>TGS:ID_c||ID_v||$Ticket_{tgs}$
4. TGS >>C:$ticket_v$

Onceper servicesession:

5. C>>V:ID_c||$Ticket_v$

$Ticket_{tgs}$=
Ek_{tgs}(ID_c||AD_c||IDt_{gs}||TS_1||$Lifetime_1$)$Ticket_v$=
Ek_v(ID_c||AD_c||ID_v||TS2||$Lifetime_2$)

C:Client, AS:Authentication Server, V:Server,
IDc: IDof theclient, Pc:Passwordoftheclient,
ADc:Addressofclient,IDv :IDoftheserver, Kv:secret key sharedby ASand V,
|| : concatenation, IDtgs:IDoftheTGSserver,TS1,TS2:timestamps,
lifetime:lifetimeoftheticket.

Thenew service,TGS,issuestickets to users whohavebeenauthenticated toAS.Thus, the user first requests a ticket-granting ticket ($Ticket_{tgs}$) from the AS. The client module intheuserworkstationsavesthis ticket.

Each time the user requires access to a new service, the client applies to the TGS, usingthe ticket to authenticate itself. The TGS then grants a ticket for the particular service. The clientsaves each service-granting ticket and uses it to authenticate its user to a server each time aparticularserviceisrequested.

Letuslookat thedetailsofthisscheme:

1. The client requests a ticket-granting ticket on behalf of the user by sending its user's IDand password to the AS, together with the TGS ID, indicating a request to use the TGSservice
2. The AS responds with a ticket that is encrypted with a key that is derived from the user'spassword.

When this response arrives at the client, the client prompts the user for his or herpassword,generates thekey,and attempts to decryptthe incomingmessage.

Ifthecorrectpasswordissupplied, theticket issuccessfullyrecovered.

Because only the correct user should know the password, only the correct user canrecovertheticket.Thus,wehaveusedthepasswordtoobtaincredentialsfromKerberoswithout having to transmit the password in plaintext. Now that the client has a ticket-grantingticket,accesstoany servercanbeobtainedwithsteps3 and4:

3. The client requests a service-granting ticket on behalf of the user. For this purpose, theclient transmits a message to the TGS containing the user's ID, the ID of the desiredservice,andthe ticket-grantingticket
4. The TGS decrypts the incoming ticket and verifies the success of the decryption by thepresence of its ID. It checks to make sure that the lifetime has not expired. Then itcompares the user ID and network address with the incoming information to authenticatethe user. If the user is permitted access to the server V, the TGS issues a ticket to grantaccess tothe requestedservice.

The service-granting ticket has the same structure as the ticket-granting ticket. Indeed,becausetheTGSisaserver,wewouldexpectthatthesameelementsareneededtoauthenticate aclienttothe TGSandtoauthenticateaclienttoanapplicationserver.

Again, the ticket contains a timestamp and lifetime. If the user wants access to the sameservice at a later time, the client can simply use the previously acquired service-granting ticketandneednotbothertheuserforapassword.

Note that the ticket is encrypted with a secret key (K_v) known only to the TGS and theserver,preventingalteration.

Finally,withaparticularservice-grantingticket,theclientcangainaccesstothecorrespondingservicewithstep5:

5. The client requests access to a service on behalf of the user. For this purpose, the clienttransmitsa messageto theservercontainingtheuser'sID and the service-grantingticket.The serverauthenticatesbyusingthe contentsoftheticket.

This new scenario satisfies the two requirements of only one password query per usersessionandprotectionofthe userpassword.

KerberosV4AuthenticationDialogueMessageExchange

Twoadditional problemsremain inthemore secureauthentication dialogue:

- Lifetime associated with the ticket granting ticket. If the lifetime is very short, thenthe user will be repeatedly asked for a password. If the lifetime is long, then theopponenthasthegreateropportunityforreplay.

- Requirementfor theservers toauthenticatethemselvestousers.

Theactual Kerberosprotocolversion 4isasfollows

:

- Abasic third-partyauthenticationscheme
- Have anAuthenticationServer(AS)
  - UsersinitiallynegotiatewithAStoidentifyself
  - AS provides a non-corruptible authentication credential (ticket grantingticketTGT)

- HaveaTicketGranting
  - Userssubsequently requestaccessto otherservicesfromTGSonbasisofusersTGT

(a) Authentication serviceexchange:toobtainticketgranting ticket
(1) C→AS :ID_C II ID_{tgs} II TS_1 (2) AS →C:EKc[$K_{c,tgs}$ II ID_{tgs} II TS_2 II $Lifetime_2$ II $Ticket_{tgs}$]
(b)Ticket-Granting ServiceExchange:toobtainservice-grantingticket

(3) $C \rightarrow TGS: ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) $TGS \rightarrow C: E_{K_{c,tgs}}[K_{c,v} \| ID_v \| TS_4 \| Ticket_v]$

$Ticket_{tgs} = E_{K,tgs}[K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2]$ $Ticket_v = E_{K_v}[K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4]$ $Authenticator_C = E_{K_{tgs}}[ID_C \| AD_C \| TS_3]$

(c)Client/Server AuthenticationExchange:toobtainservice

(5) $C \rightarrow V: Ticket_v \| Authenticator_c$

(6) $V \rightarrow C: E_{k_{c,v}}[TS_5+1]$

$Ticket_v = E_{K_v}[K_{c,v} \| ID_C \| AD_C \| Id_v \| TS_4 \| Lifetime_4]$

$Authenticator_c = E_{K_{tgs}}[ID_C \| AD_C \| TS_3]$

Kerberos4Overview

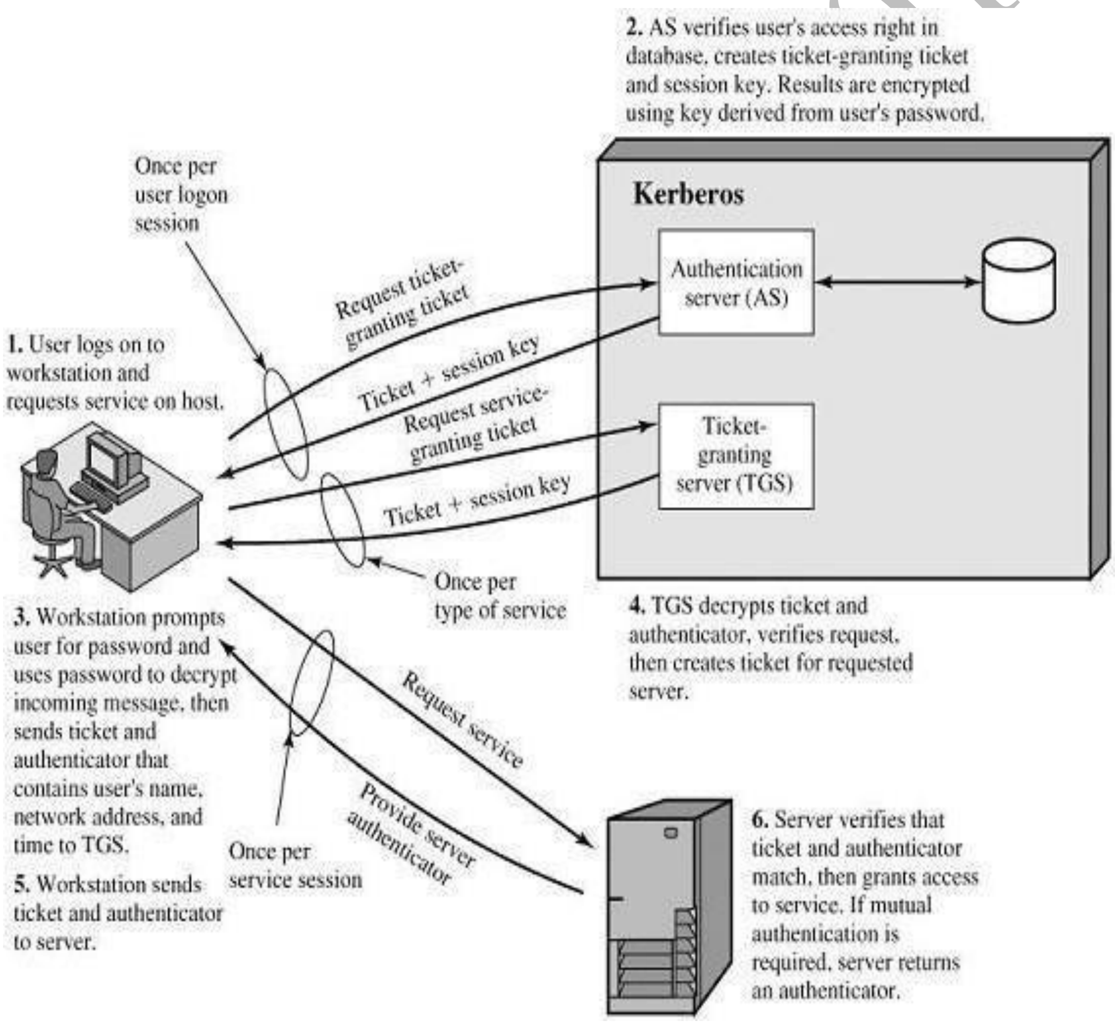


Fig4.1OverviewofKerberos4

KerberosRealmsandMultipleKerberi

Afull-serviceKerberos environmentconsisting ofaKerberosserver,anumber ofclients,andanumberofapplicationserversrequires thefollowing:

4. TheKerberosserver musthave theuser ID andhashed passwordsofallparticipatingusersinitsdatabase.All usersare registeredwiththeKerberosserver.
5. TheKerberosserver mustshare a secretkeywitheach server.All serversareregisteredwiththe Kerberosserver.

Suchanenvironmentis referredtoas

a**KerberosrealmTheconceptof*realm*can**

beexplainedasfollows.

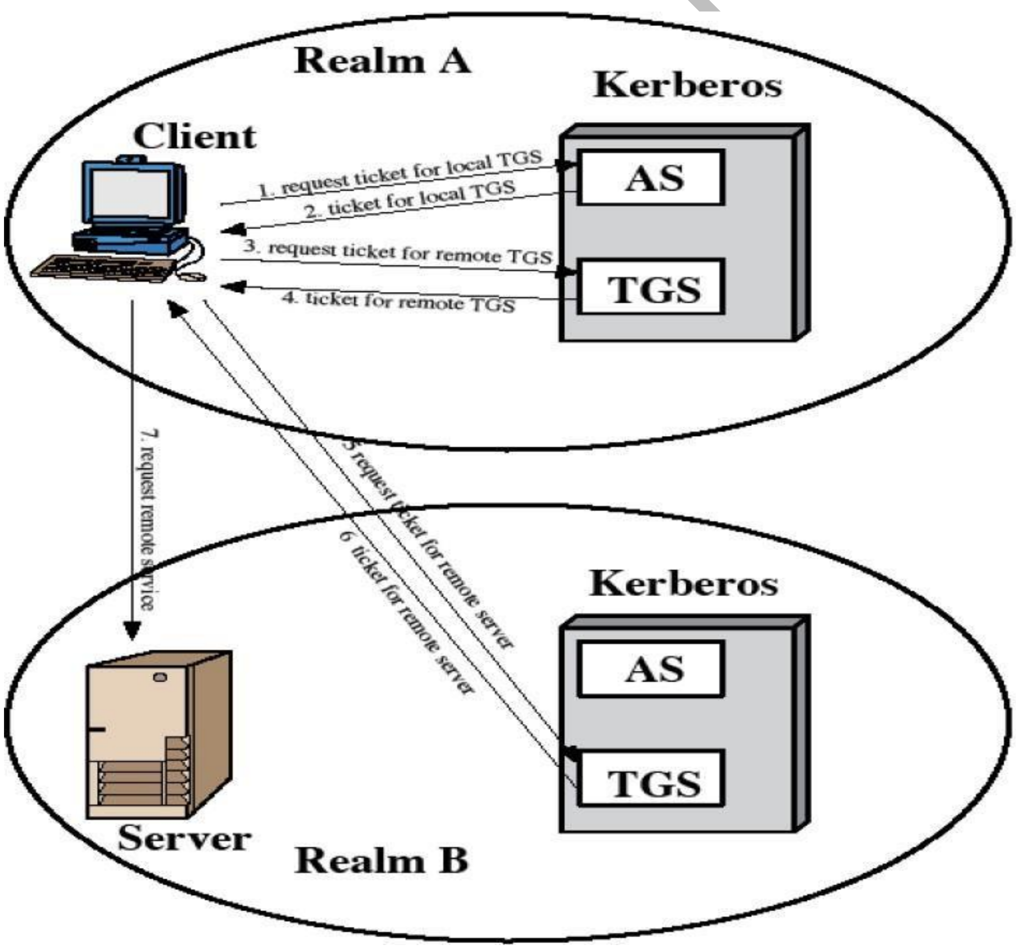


Fig.Requestforservice inanotherRealm

A Kerberos realm is a set of managed nodes that share the same Kerberos database.The Kerberos database resides on the Kerberos master computer system, which should be keptin a physically secure room. A read-only copy of the Kerberos database might also reside onotherKerberoscomputersystems.

However, all changes to the database must be made on the master computer system.Changing or accessing the contents of aKerberos database requires the Kerberos masterpassword. A related concept is that of a Kerberos principal, which is a service or user that isknownto the Kerberossystem.

Each Kerberos principal is identified by its principal name. Principal names consist ofthree parts: a service or user name, an instance name, and a realm name. Networks of clientsandserversunderdifferentadministrativeorganizations typicallyconstitutedifferent realms.

That is, it generally is not practical, or does not conform to administrative policy, to haveusersandserversin oneadministrativedomainregisteredwith aKerberosserver elsewhere.

However, users in one realm may need access to servers in other realms, and someservers may be willing to provide service to users from other realms, provided that those usersareauthenticated.

Kerberos provides a mechanism for supporting such inter realm authentication. For tworealms tosupportinterrealmauthentication,athirdrequirementisadded:

6. The Kerberos server in each interoperatingrealmsharesasecretkey with theserverin theotherrealm.The twoKerberos serversare registeredwitheachother.

The scheme requires that the Kerberos server in one realm trust the Kerberos server inthe other realm to authenticate its users. Furthermore, the participating servers in the secondrealmmustalsobewillingto trustthe Kerberos serverinthefirstrealm.

Thedetailsoftheexchanges illustratedin Fig2are asfollows:

C→AS :$ID_C \,||\, ID_{tgs} \,||\, TS_1$

AS→C :$E K_c[K_{c,tgs} \,||\, ID_{tgs} \,||\, TS_2 \,||\, Lifetime_2 \,||\, Ticket_{tgs}$ C→

TGS :$ID_{tgsrem} \,||\, Ticket_{tgs} \,||\, Authenticator_c$

TGS→C :$E\ K_{c,tgs}[K_{c,tgsrem} \,||\, ID_{tgsrem} \,||\, TS_4 \,||$

$Ticket_{tgsrem}$ C→ TGS $_{rem}$:$ID_{vrem} \,||$

$Ticket_{tgsrem} \,||\, Authenticator_c$

TGS $_{rem}$→C:$E K_{c,tgsrem}[\, K_{c,vrem} \,||\, ID_{vrem} \,||\, TS_6 \,||\, Ticket_{vrem}$:

C→V_{rem} :$Ticket_{vrem} \,||\, Authenticator_c$

DifferencesbetweenVersions4and5

Version 5 is intended to address the limitations of version 4 in two areas: environmentalshortcomingsand technicaldeficiencies.

Environmentalshortcomings:

7. **Encryptionsystemdependence:**

Version 4 requires the use of DES. In version 5, ciphertext is tagged with an encryptiontypeidentifiersothatanyencryptiontechnique may beused.

8. **Internetprotocoldependence:**

Version4requirestheuseofInternetProtocol(IP)addresses.Version5networkaddresses are taggedwith typeandlength,allowingany networkaddress typetobe used.

9. **Messagebyte ordering:**

In version 4, the sender of a message employs a byte ordering of its own choosing andtags the message to indicate least significant byte in lowest address In version 5, all messagestructures are defined using Abstract Syntax Notation One (ASN.1) and Basic Encoding Rules(BER),whichprovideanunambiguousbyteordering.

10. **Ticketlifetime:**

Lifetime values in version 4 are encoded in an 8-bit quantity in units of five minutes. Inversion 5, tickets include an explicit start time and end time, allowing tickets with arbitrarylifetimes.

11. **Authenticationforwarding:**

Version 4 does not allow credentials issued to one client to be forwarded to some otherhostand usedby someotherclient.Version5 providesthiscapability.

Technicaldeficienciesintheversion4protocol:

- Doubleencryption
- PCBC encryption
- Sessionkeys
- Passwordattacks

TheVersion5 AuthenticationDialogue

(a)AuthenticationServiceExchange:toobtainticket-grantingticket
(1) C →AS:Options IIID_c II$Realm_c$ IITimesIINonce$_1$
(2) AS → C : $Realm_c$ II ID_c II $Ticket_{tgs}$ II EK_c [$K_{c,tgs}$ II Times II $Nonce_1$ II $Realm_{tgs}$ II ID_{tgs}]$Ticket_{tgs}$=EK_{tgs}[Flags II$K_{c,tgs}$II$Realm_c$ IIID_c IIAD_cIITimes]
(b)Ticket–GrantingServiceExchange:toobtain service-grantingticket
(3) C→ TGS:OptionnsIIID_vIITimes IINonce$_1$
(4) TGS → C : $Realm_c$ II ID_c II $Ticket_v$ II $EK_{c,tgs}$[$K_{c,v}$ II Times II $Nonce_2$ II $Realm_v$ II ID_v]$Ticket_{tgs}$=EK_{tgs}[FlagsII$K_{c,tgs}$II$Realm_c$IIID_c IIAD_cIITimes]
$Ticket_v$=Ek_v[[FlagsII$K_{c,v}$II$Realm_c$IIID_cIIAD_cIITimes]

Authenticator$_c$=EK$_{c,tgs}$[ID$_c$II Realm$_c$IITS$_1$]

(c)Client/Server AUTHENTICATIONExchange:toobtainservice

(5) C →V:Options IITicket$_v$IIAuthenticator$_c$
(6) V→C:EK$_{c,v}$[TS$_2$ IIsubkeyIISeq#]
Ticketv =EK$_v$[FlagsIIK$_{c,v}$IIRealm$_c$ IIID$_c$IIAD$_c$
IITimes]Authenticator$_c$=E$_{Kc,v}$[ID$_c$II Realm$_c$IITS$_2$IISubkeyIISeq#]

First, consider the authentication service exchange. Message (1) is a client request for aticket-grantingticket.Itincludesthe IDoftheuser andtheTGS.

Thefollowingnewelementsare added:

- Realm:Indicatesrealmofuser
- Options:Usedtorequestthatcertainflagsbesetinthereturnedticket
- Times:Usedbytheclienttorequestthefollowingtimesettingsintheticket:
  - from :thedesiredstarttimefortherequestedticket
  - till : therequestedexpirationtimefortherequestedticket
  - r_{time} :requestedrenew-tilltime

Nonce:Arandomvaluetobe repeatedin message (2)toassure thattheresponseisfreshandhasnotbeen replacedbyanopponent.

Message (2) returns a ticket-granting ticket, identifying information for the client, and ablock encrypted using the encryption key based on the user's password. This block includes thesession key to be used between the client and the TGS, times specified in message (1), thenoncefrommessage(1),andTGSidentifyinginformation.

Theticketitselfincludesthesessionkey,identifyinginformationfortheclient,therequestedtime values,andflags that reflectthestatus ofthis ticketandtherequestedoptions.

Letusnow comparethe ticket-grantingserviceexchangeforversions 4and5.

We see that message (3) for both versions include an authenticator, a ticket, and thenameofthe requestedservice.

Inaddition,version5includesrequestedtimesandoptionsfortheticketandanonce,all with functions similar to those of message (1). The authenticator itself is essentially the sameasthe oneusedinversion4.

Theauthenticatoritselfis essentiallythesameas theoneused inversion 4.

Message (4) has the same structure as message (2), returning a ticket plus informationneeded by the client, the latter encrypted with the session key now shared by the client and theTGS.

Finally, for the client/server authentication exchange, several new features appear inversion 5. In message (5), the client may request as an option that mutual authentication isrequired.Theauthenticatorincludesseveralnewfieldsas follows:

- **Subkey**: The client's choice for an encryption key to be used to protect this specificapplicationsession.Ifthisfieldisomitted,thesession keyfromthe ticket(Kc,v)isused.

- **Sequence number**: An optional field that specifies the starting sequence number to beusedbytheserverformessagessenttotheclientduringthissession.Messagesmaybesequence numbered todetectreplays.

Ifmutualauthenticationisrequired,theserverrespondswithmessage(6).Thismessage includes the timestamp from the authenticator. Note that in version 4, the timestampwas incremented by one. This is not necessary in version 5 because the nature of the format ofmessagesissuchthatitisnotpossibleforanopponenttocreatemessage(6)withoutknowledgeoftheappropriateencryption keys.

X.509AUTHENTICATIONSERVICES

X.509 defines a framework for authentication services by the X.500 directory to its users.Thedirectoryconsistsofpublic-key certificates.

Each certificate contains the public key of a user and is signed with the private key of atrustedcertificationauthority.

X.509 defines authentication protocols based on public key certificates. X.509 standardcertificate formatused inS/MIME,IPSecurityandSSL/TLSand SET.

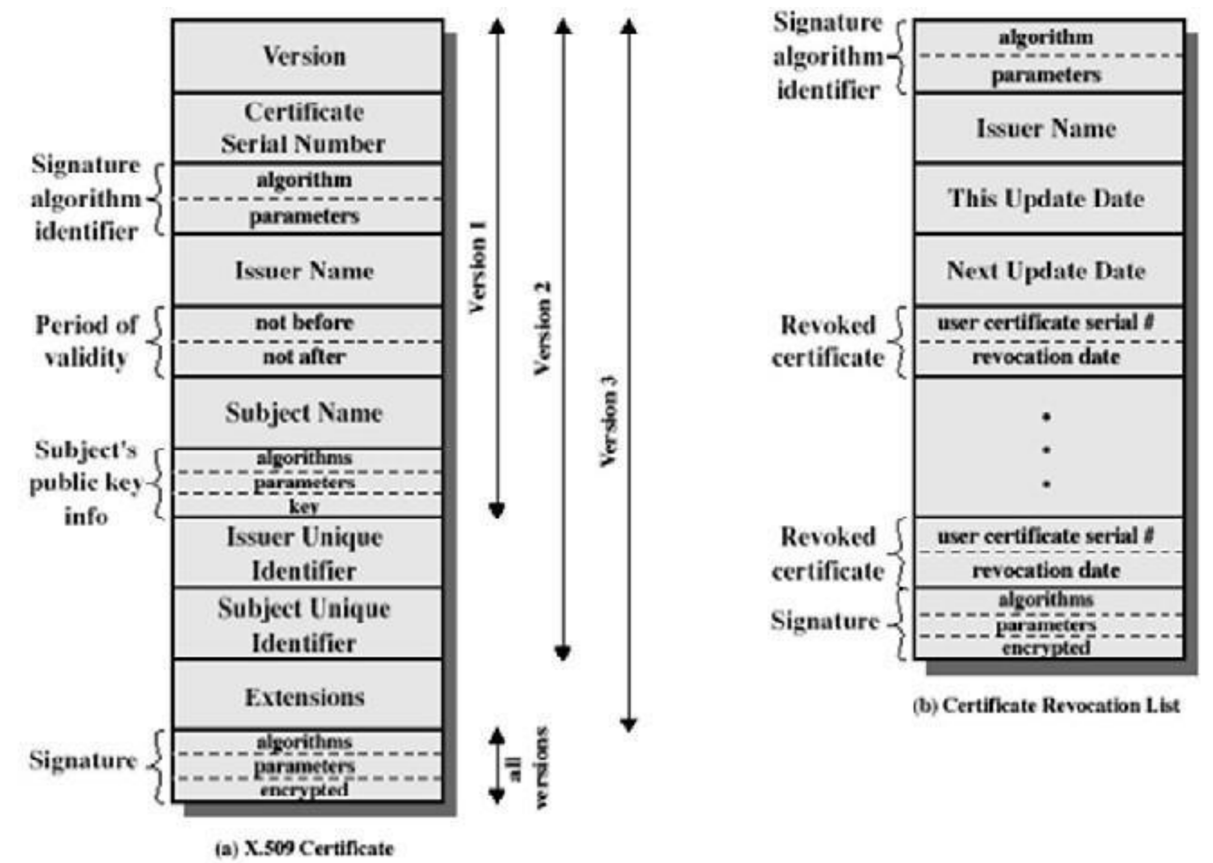
Certificates

ThecertificatesarecreatedandstoredinthedirectorybythetrustedCertificationAuthority (CA). The directory server not having certification functions and not create public key.Butthe userobtainsthecertificatefromsomeeasily accessiblelocation

The general format of the certificate as shown below Fig

4.3Theelementsofthecertificates are

1. **Version(V):** The default version is 1. The issuer and subject unique identifier are presentinversion2.Ifoneormoreextensionsarepresentinversion3.
2. **SerialNumber(SN):**UniqueintegervalueissuedbyCA
3. **Signature Algorithm Identifier (AI):** This algorithm is used to sign the certificate withsomeparameters
4. **IssuerName(CA):**Thename of theCAthatcreated andsignedthiscertificate
5. **Periodofvalidity(T_A):**Thefirstandlastonwhich thecertificateisvalid
6. **SubjectName(A):**Thenameof theuser towhomthiscertificaterefers
7. **Subject's Public Key Information (AP):** The public key of the subject plus identifier ofthealgorithmforwhichthis key istobeused,withassociatedparameters.



FigX.509AUTHENTICATIONSERVICES

IssuerUniqueIdentifier: It isusedtoidentifyuniquelytheissuing CA

8. **SubjectUniqueIdentifier:** It isusedto identify uniquelythesubject
9. **Extensions:**A set ofone or moreextensionfields
10. **Signature:**Coversall of the otherfieldsofcertificate;itcontainshash codeofotherfields,encryptedwiththeCA‟sprivatekey.

[**Note:**Uniqueidentifier isused toavoidreuse ofsubjectandissuernamesovertime]

Notationtodefineacertificate

CA<<A>> =CA{V,SN,AI,CA, TA,A,Ap}
where
Y<<X>>=Thecertificate ofuserXissuedbycertificationauthority Y.
Y {I}=ThesigningofIby Y.Itconsists ofIwith anencryptedhashcodeappended.

The CA signs the certificate with its private key. If the corresponding public key is knowntoauser,thenthatusercanverifythata certificatesignedbytheCAisvalid.

Generationandusageof certificate byauser

Theuser certificatesgeneratedbyCA have thefollowingcharacteristics:

11. Anyuserwithaccess tothepublic keyoftheCAcanverifytheuserpublickeythatwascertified.
12. No party other than the Certification Authority (CA) can modify the certificate withoutthisbeingdetected.
13. Certificatesareunforgeable,

If all users belong to the same CA, the certificates can be placed in the directory foraccess by all users. If the number of users increased, single CA could not be satisfied all userrequirements.

For example, User A has obtained the certificate from certificate authority x_1 and user Bfrom $x_{2.}$ Now the two CAs (x_1 and x2) securities exchange their own public keys in the form ofcertificates.Thatis$x_1$mustholdx_2‟scertificateand$x_2$holdsx_1‟scertificate

NowAwantstoaccessB‟s publickey,itfollowsthefollowingchaintoobtainB‟spublic key.

$x_1<<x_2>>x_2<<B>>$

i.e.,first Agetsx_2‟scertificatefromx_1‟sdirectorytoobtain x_2‟spublickey.Thenusing x_2‟spublickeytoobtainB‟scertificatefrom x_2‟sdirectorytoobtain„spublickey.

In the same method, B can obtain A‟s public key with the reverse chain$x_2<<x_1>>x_1<<A>>$

HierarchyofCAs

To obtain public key certificate of user efficiently, more than one CAs can be arranged inahierarchy,sothatnavigationineasy.

The connected circles indicate the hierarchical(Fig 4.4) relationship among the CAs; theassociatedboxesindicatecertificatesmaintainedinthedirectoryforeachCAentry.Thedirectory entryforeachCAincludestwotypesofcertificates:

- **Forwardcertificates**:Certificates ofXgeneratedbyotherCAs
- **Reverse certificates**: Certificates generated by X that are the certificatesof otherCAs

User A can acquire the following certificates from the directory to establish a certificationpathtoB:

X<<W>>W<<V>>V <<Y>><<Z>>Z <<B>>

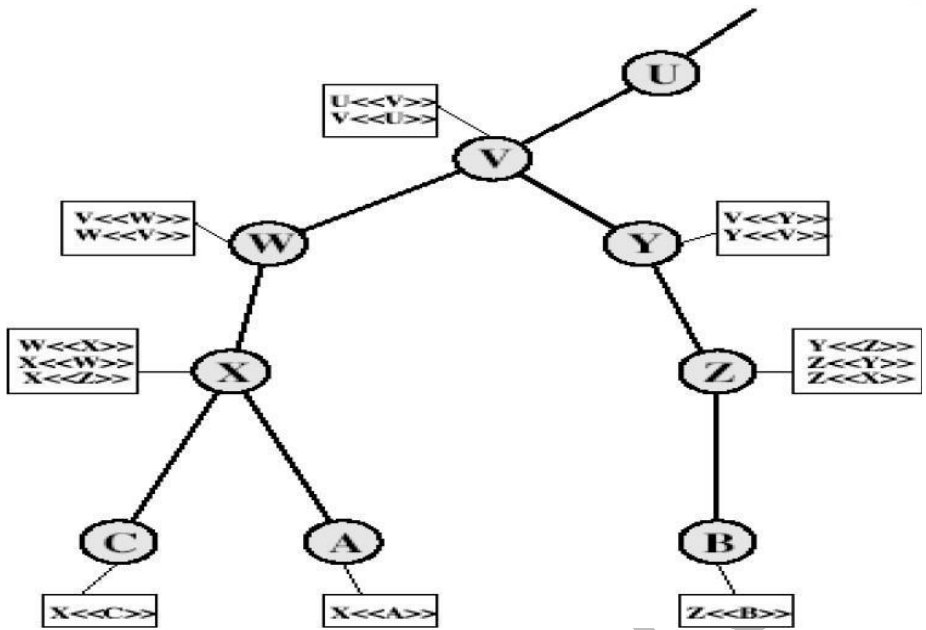


Fig4.4:HierarchyofX.509

Revocationofcertificates

- Certificateshaveaperiod ofvalidity
- Mayneedtorevoke beforeexpiry, forthefollowingreasonseg:
  - ✓ User's private keyiscompromised
  - ✓ Userisnolongercertified bythisCA
  - ✓ CA'scertificateiscompromised
- CA‟smaintainlist ofrevokedcertificates
  - ✓ TheCertificateRevocation List(CRL)
- Users shouldcheckCertificateswithCA‟sCRL

AuthenticationProcedures

X.509includesthreealternativeauthenticationprocedures:

- One-WayAuthentication
- Two-WayAuthentication
- Three-WayAuthentication

One-WayAuthentication

Onemessage(A→B) usedtoestablish

- TheidentityofAandthatmessageis fromA
- Messagewasintended forB
- Integrity&originalityofmessage

Messagemustinclude timestamp,nonce,B'sidentity and is signedbyA

Two-WayAuthentication

Twomessages(A→B,B→A)whichalsoestablishesinaddition:

- TheidentityofBandthatreply is fromB
- That reply isintended forA
- Integrity&originalityofreply

Replyincludesoriginalnoncefrom A,also timestampandnoncefrom B

Three-WayAuthentication

Three messages (A→B, B→A, A→B) which enables above authentication withoutsynchronizedclocks(Fig4.5)

- HasreplyfromAbacktoBcontainingsignedcopyofnoncefromB
- Means that timestampsneednotbecheckedor reliedupon

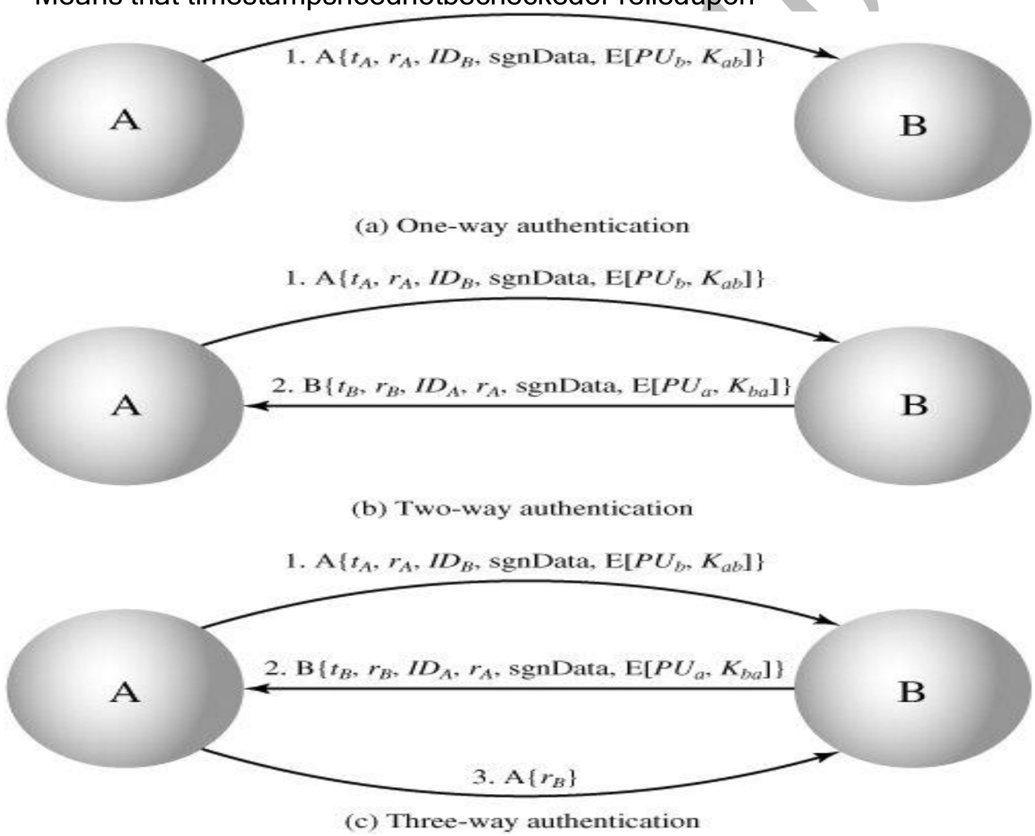


Fig:X509StrongAuthenticationProcedure

X.509 Version3

Thefollowingrequirementsnotsatisfiedbyversion2:

14. The Subject field is inadequate to convey the identity of a key owner to a public-keyuser.
15. TheSubjectfieldisalsoinadequateformanyapplications,whichtypicallyrecognize entities by an Internet e-mail address, a URL, or some other Internet-relatedidentification.
16. There is a need to indicate security policy information. There is a need to limit thedamage that can result from a faulty or malicious CA by setting constraints on theapplicability ofaparticularcertificate.
17. It is important to be able to identify different keys used by the same owner atdifferent times.

The certificate extensions fall into three main categories: key and policy information,subjectand issuerattributes,andcertificationpath constraints.

KeyandPolicyInformation

These extensions convey additional information about the subject and issuer keys, plusindicators of certificate policy.. For example, a policy might be applicable to the authentication ofelectronic data interchange (EDI) transactions for the tradingof goods within a given pricerange.

This area includesthefollowing:

Authority key identifier: Identifies the public key to be used to verify the signature onthiscertificateorCRL.
Subjectkeyidentifier: Identifiesthe publickeybeingcertified.
Key usage: Indicatesarestrictionimposedastothepurposesforwhich,andthepoliciesunderwhich, thecertifiedpublickeymaybeused.
Private-key usage period: Indicates the period of use of the private key correspondingto the public key. For example, with digital signature keys, the usage period for thesigningprivate keyistypically shorterthan thatfor theverifyingpublic key.
Certificate policies: Certificates may be used in environments where multiple policiesapply.
Policymappings: Usedonlyincertificates for CAsissuedby otherCAs.

CertificateSubjectandIssuerAttributes

Theseextensionssupportalternativenames,inalternativeformats,foracertificatesubject or certificate issuer and can convey additional information about thecertificate subject,to increase a certificate user's confidence that the certificate subject is a particular person orentity. For example, information such as postal address, position within a corporation, or pictureimagemay berequired.

Theextensionfields inthisareaincludethefollowing:

- **Subjectalternativename:** Contains oneormorealternative names, usinganyofavariety of forms

- **Subject directory attributes:** Conveys any desired X.500 directory attribute valuesfor the subjectofthiscertificate.

CertificationPathConstraints

TheseextensionsallowconstraintspecificationstobeincludedincertificatesissuedforCAsby otherCAs. Theextension fieldsinthisareaincludethefollowing:

- **Basicconstraints:**Indicatesifthe subject mayactasaCA. Ifso,acertificationpathlength constraintmay bespecified.
- **Name constraints**: Indicates a name space within which all subject names insubsequentcertificates inacertificationpath mustbelocated.
- **Policy constraints**: Specifies constraints that may require explicit certificate policyidentification orinhibitpolicymappingfortheremainderofthecertificationpath.

UNITV-SECURITYPRACTICEANDSYSTEMSECURITY

ElectronicMailsecurity–PGP,S/MIME–IPsecurity–WebSecurity-SYSTEMSECURITY:Intruders–Malicious software– viruses –Firewalls.

ElectronicMailsecurity

5.1.1 PRETTYGOODPRIVACY(PGP)

PGPprovidestheconfidentialityandauthenticationservicethatcanbeusedforelectronicmailand filestorageapplications.

Thesteps involvedin PGP are

□ Selectthebestavailablecryptographicalgorithmsasbuildingblocks.
□ Integratethesealgorithmsintoageneralpurposeapplicationthatisindependentofoperatingsystemand processorand that is based on asmall set ofeasy-tousecommands.
□ Makethepackageanditsdocumentation,includingthesourcecode,freelyavailableviatheinternet,bulletin boards and commercial networks.
□ Enterintoanagreementwithacompanytoprovideafullycompatible,lowcostcommercialversion ofPGP.

PGPhasgrown explosivelyandis nowwidelyused.

Anumberofreasonscanbecitedforthisgrowth.
□ It isavailablefreeworldwide in versions that run on avarietyof platform.
□ Itisbasedonalgorithmsthathavesurvivedextensivepublicreviewandareconsideredextremelysecure.e.g., RSA, DSS and DiffieHellman forpublickeyencryption
□ Ithasawiderangeofapplicability.
□ Itwasnotdeveloped by,norit iscontrolledby,anygovernmentalorstandardsorganization.

5.1.1.1. Operationaldescription

Theactualoperation ofPGP consistsof fiveservices:
1. Authentication
2. Confidentiality
3. Compression
4. E-mail compatibility
5. Segmentation.

1. Authentication: Thesequencefor authenticationis asfollows:

□ Thesender createsthe message

□ SHA-1isusedtogeneratea160-bithashcodeofthemessage

□ Thehashcodeisencryptedwith RSAusingthesender‟sprivatekeyandtheresultispretendedto themessage.

- The receiver uses RSA with the sender‟s public key to decrypt and recover thehashcode.
- The receiver generates a new hash code for the message and compares it withthedecryptedhashcode.Ifthetwomatch,themessageisacceptedasauthentic.

2. Confidentiality

Confidentiality is provided by encrypting messages to be transmitted or to be storedlocallyas files.Inbothcases, theconventionalencryptionalgorithmCAST-128 maybeused.

The 64-bit cipher feedback (CFB) mode is used. In PGP, each conventional key is usedonly once. That is, a new key is generated as a random 128-bit number for each message. Thusalthough this is referred to as **a session key**, it is in reality a **onetime key**. To protect the key, itisencryptedwiththereceiver‟spublickey.
Thesequenceforconfidentialityisasfollows:

- The sender generates a message and a random 128-bit number to be used as asessionkeyforthismessage only.
- Themessageis encryptedusingCAST-128 with thesessionkey.
- ThesessionkeyisencryptedwithRSA,usingthereceiver‟spublickeyandisprependedt othe message.
- ThereceiverusesRSAwithitsprivatekeytodecryptandrecoverthesessionkey.
- Thesession keyisused todecryptthemessage.

Confidentialityandauthentication

Here both services may be used for the same message. First, a signature is generatedfor the plaintext message and prepended to the message. Then the plaintext plus the signatureisencryptedusingCAST-128andthe session keyisencryptedusingRSA.

3. Compression

As a default, PGP compresses the message after applying the signature but beforeencryption. Thishasthebenefitof savingspaceforbothe-mailtransmissionandforfile storage.

Thesignature isgeneratedbefore compressionfor two reasons:

- It is preferable to sign an uncompressed message so that one can store only theuncompressed message together with the signature for future verification. If onesignedacompresseddocument,thenitwouldbenecessaryeithertostoreacompressed versionofthemessageforlaterverificationortorecompressthemessagewhenverificationi srequired.
- Even if one were willing to generate dynamically a recompressed message fromverification, PGP‟s compression algorithm presents a difficulty. The algorithm is notdeterministic; various implementations of the algorithm achieve different tradeoffs inrunningspeedversuscompressionratioandasaresult,producedifferentcompressionfor ms.

Message encryption is applied after compression to strengthen cryptographic security. Becausethe compressed message has less redundancy than the original plaintext, cryptanalysis is moredifficult.Thecompressionalgorithmused isZIP

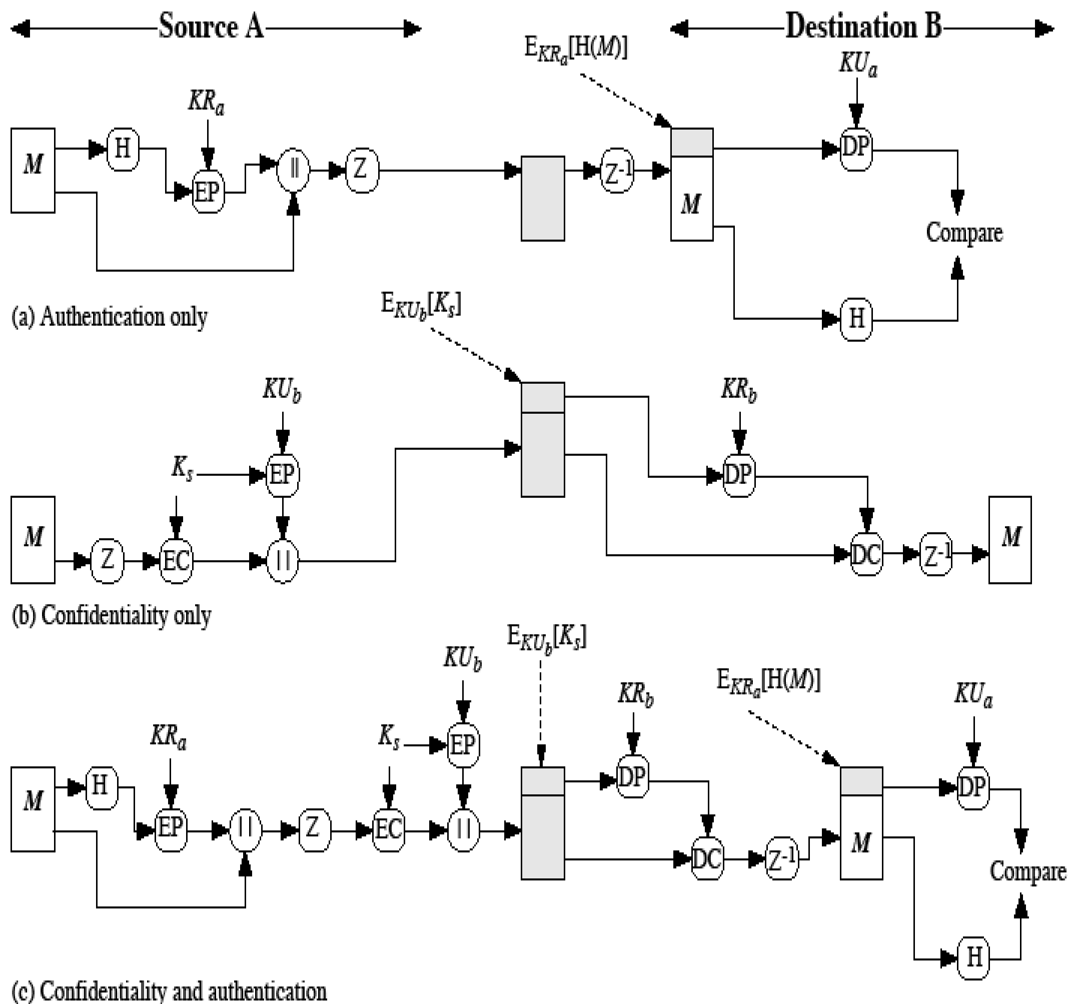


Fig5.1:PGPCryptographicFunctions

4. E-mailcompatibility

Many electronic mail systems only permit the use of blocks consisting of ASCII texts. Toaccommodatethisrestriction,PGPprovidestheserviceofconvertingtheraw8-bitbinarystream to a stream of printable ASCII characters. The scheme used for this purpose is **radix-64conversion**.Eachgroupofthreeoctetsofbinarydatais mappedinto four ASCIIcharacters.

5. Segmentationandreassembly

E-mail facilities often are restricted to a maximum length. E.g., many of the facilitiesaccessiblethrough the internetimposea maximumlength of50,000 octets.Any messagelongerthanthatmustbebrokenupintosmaller segments,each ofwhichis mailedseparately.

To accommodate this restriction, PGP automatically subdivides a message that is toolarge into segments that are small enough to send via e-mail. The segmentation is done after alltheotherprocessing,includingthe radix-64conversion.

At the receiving end, PGP must strip off all e-mail headers and reassemble the entireoriginalblockbeforeperformingtheothersteps.

5.1.1.2. Cryptographickeys andkeyrings

Threeseparate requirementscanbeidentifiedwith respecttothesekeys:

- Ameansofgeneratingunpredictable sessionkeysis needed.
- It mustallowa user tohave multiple publickey/privatekeypairs.
- EachPGP entitymustmaintaina fileofitsownpublic/privatekeypairsaswellasa fileofpublickeysofcorrespondents.

a. Sessionkeygeneration

Eachsessionkeyisassociatedwithasinglemessageandisusedonlyforthepurposeof encryption and decryption of that message. Random 128-bit numbers are generated usingCAST-128itself.

Theinputtotherandomnumbergeneratorconsistsofa128-bitkeyandtwo64-bitblocksthataretreated asplaintexttobeencrypted.Usingcipherfeedback mode, theCAST-128produces two64-bitciphertextblocks,whichare concatenated to formthe 128-bitsessionkey.TheplaintextinputtoCAST-128isitselfderivedfromastreamof128-bitrandomizednumbers.Thesenumbersarebasedonthe keystrokeinputfromtheuser.

b. Keyidentifiers

Ifmultiple public/private key pair are used, then how does the recipient know which ofthepublickeyswasused to encryptthe session key?

One simple solution would be to transmit the public key with the message but, it isunnecessary wasteful of space. Another solution would be to associate an identifier with eachpublickey thatisuniqueatleastwithineachuser.

The solution adopted by PGP is to assign a key ID to each public key that is, with veryhigh probability, unique within a user ID. The key ID associated with each public key consists ofitsleastsignificant64bits.i.e.,thekey IDofpublic key $KU_a$is(KU_amod2^{64}).

A messageconsistsofthreecomponents.

- **Messagecomponent**–includesactualdatatobetransmitted,aswellasthefilenameandatimestampthatspecifiesthetimeofcreation
- **Sessionkeycomponent**–includessession keyandtheidentifieroftherecipientpublickey.
- **Signaturecomponent**-includesthefollowing
  - **Timestamp**-timeatwhichthesignaturewasmade.
  - **Messagedigest**-hashcode.
  - **Twooctetsofmessagedigest**–toenabletherecipienttodetermineifthecorrectpublickeywasusedtodecryptthemessage.
  - **KeyIDofsender'spublickey**-identifiesthepublickey

Notation:

- **EkU_b**=encryptionwithuserB‟sPublickey
- **EKR_a**=encryptionwithuserA‟sprivatekey
- **EK_s**=encryption with sessionkey
- **ZIP**=Zipcompressionfunction
- **R64**=Radix-64conversion function

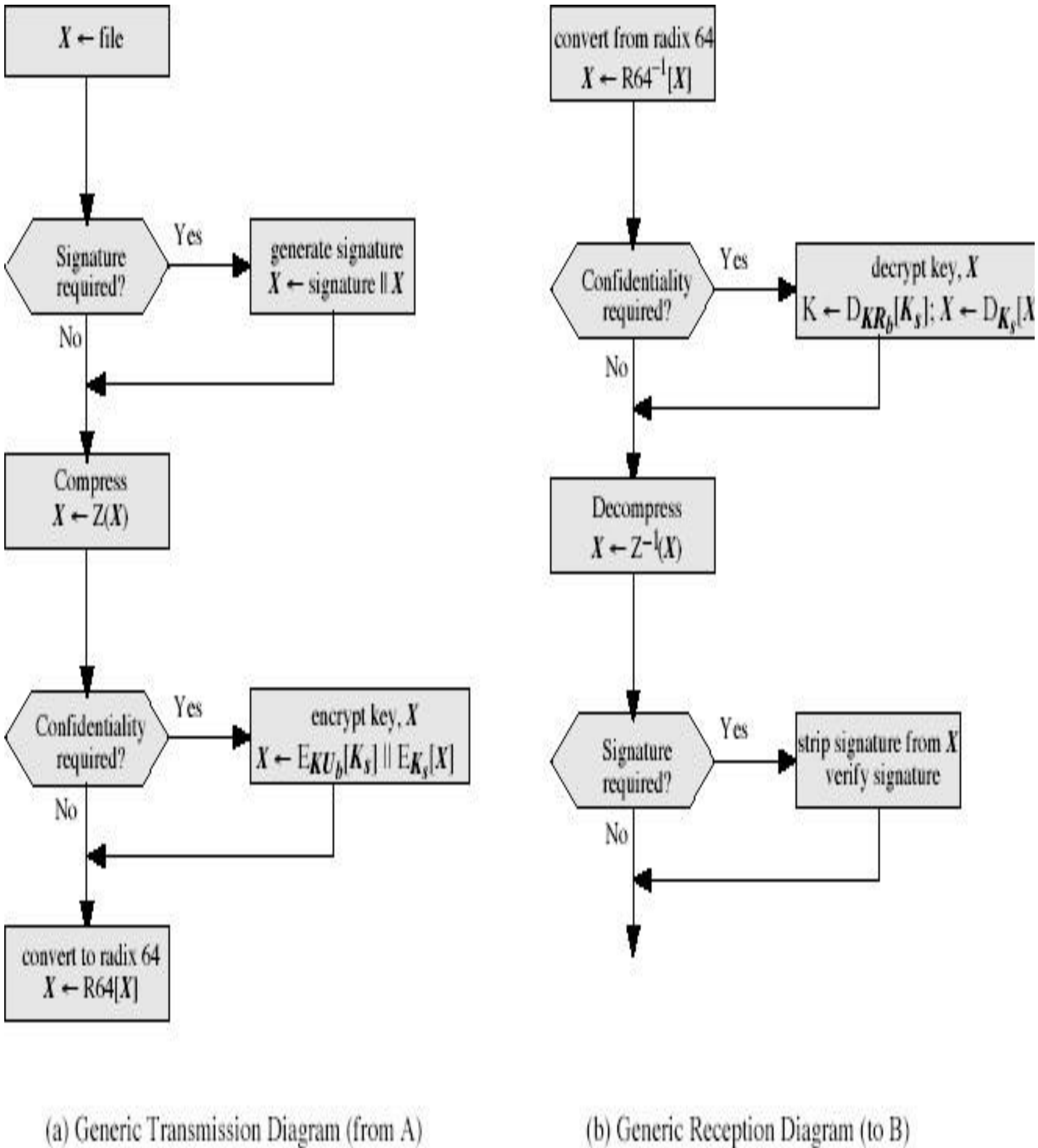


Fig5.2:TransmissionandReceptionofPGPmessage

PGP provides a pair of data structures at each node, one to store the public/private keypair owned by that node and one to store the public keys of the other users known at that node.Thesedatastructuresarereferredtoasprivatekeyringand publickey ring.

Thegeneralstructures oftheprivateandpublic keyrings are shown below:

Timestamp- the date/time when this entry was made.**KeyID -**theleast significant bitsofthepublickey.**Publickey-**publickeyportionofthepair.
Private Key-privatekey portionofthe pair.

User ID **-**theownerofthe key
Keylegitimacyfield–
indicatestheextenttowhichPGPwilltrustthatthisisavalidpublickeyforthisuser.

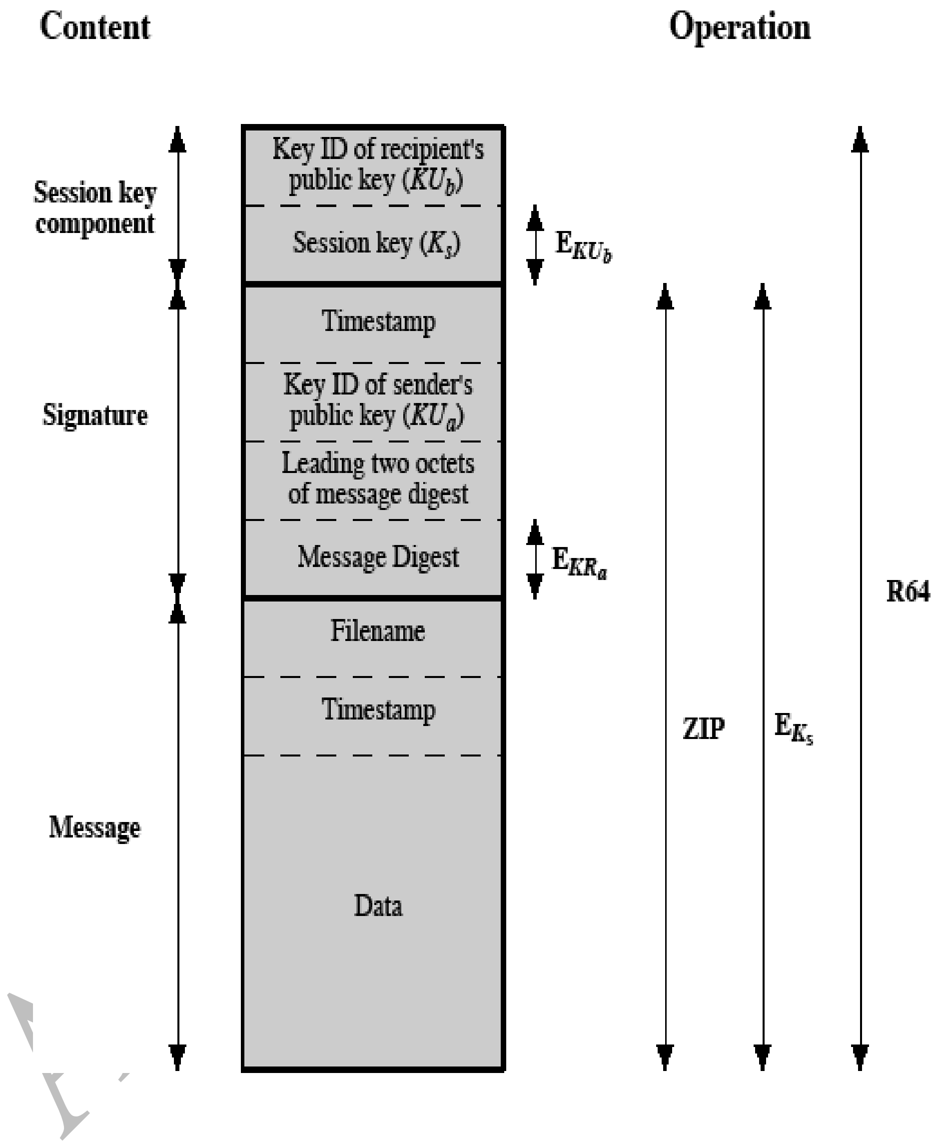


Fig5.3:GeneralFormatofPGPmessage(FromAtoB)

Signature trust field – indicates the degree to which this PGP user trusts the signer to certifypublickey.

Owner trust field - indicates the degree to which this public key is trusted to sign otherpublickeycertificates.

PGP message generation First consider message transmission and assume that themessage is to be both signed and encrypted. The sending PGP entity performs the followingsteps:

1. Signingthemessage

- PGPretrievesthesender‟sprivatekeyfromtheprivatekeyringusinguserIDasanindex.IfuserID wasnotprovided,thefirstprivate keyfromtheringis retrieved.
- PGPpromptstheuserforthepassphrase(password)torecovertheunencryptedprivatekey.
- Thesignaturecomponentof themessageisconstructed.

Private Key Ring

Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
• • •	• • •	• • •	• • •	• • •
T_i	$PU_i \bmod 2^{64}$	PU_i	$E(H(P_i), PR_i)$	User *i*
• • •	• • •	• • •	• • •	• • •

Public Key Ring

Timestamp	Key ID*	Public Key	Owner Trust	User ID*	Key Legitimacy	Signature(s)	Signature Trust(s)
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
T_i	$PU_i \bmod 2^{64}$	PU_i	$trust\_flag_i$	User *i*	$trust\_flag_i$		
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •

* = field used to index table

Fig5.4:Generalstructure ofprivateandpublickeyRings

2. Encryptingthemessage

- PGPgenerates asession key andencryptsthe message.
- PGPretrievestherecipient‟spublickeyfromthepublickeyringusinguserIDasindex.

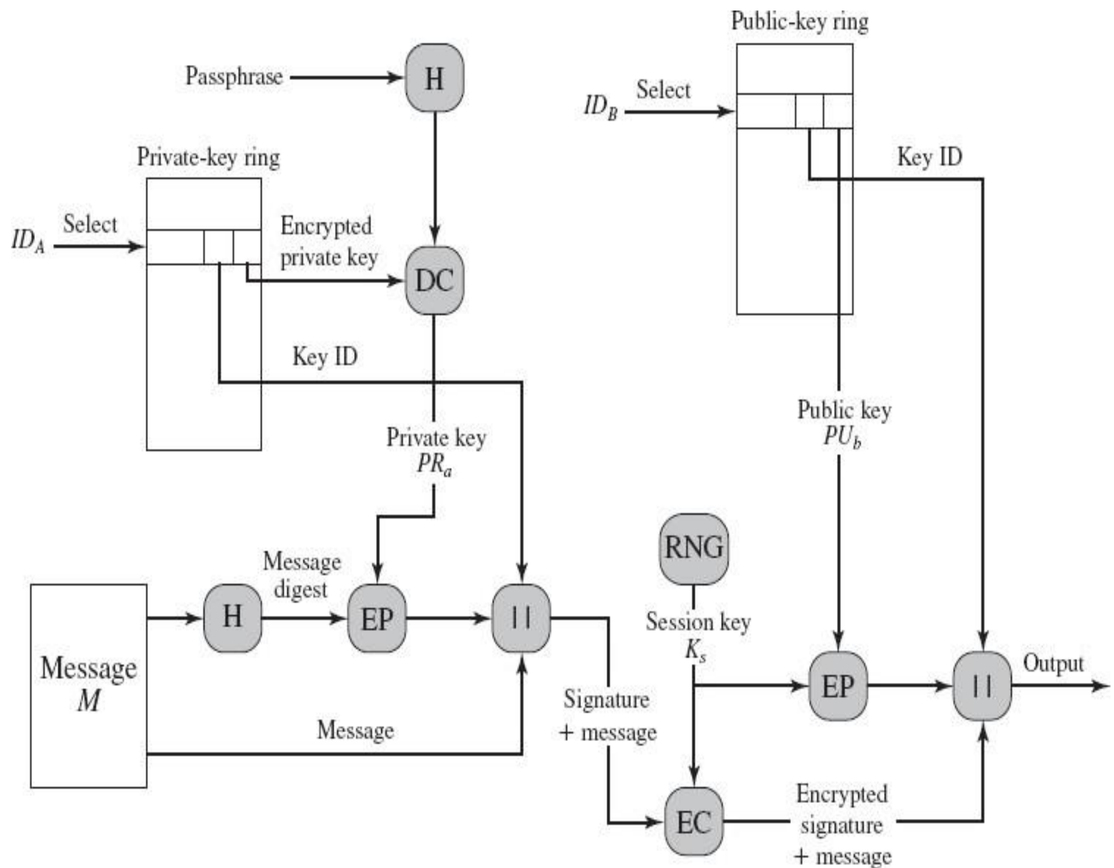


Fig5.5: PGPmessagegeneration

ThereceivingPGPentityperformsthefollowingsteps:

1. Decryptingthemessage

- PGPretrievesthereceiver‟sprivatekeyfromtheprivatekeyring,usingthekeyIDfieldinthesession key componentofthe message as anindex.
- PGPpromptstheuserforthepassphrase(password)torecovertheunencryptedprivatekey.
- PGPthen recoversthesessionkey anddecryptsthe message.

2. Authenticatingthemessage

- PGPretrievesthesender‟spublickeyfromthepublickeyring,usingthekeyIDfieldinthesignaturekey componentofthe message asanindex.
- PGP recoversthetransmittedmessage digest.
- PGPcomputesthemessagedigestforthereceivedmessageandcomparesittothetransmitted message digesttoauthenticate.

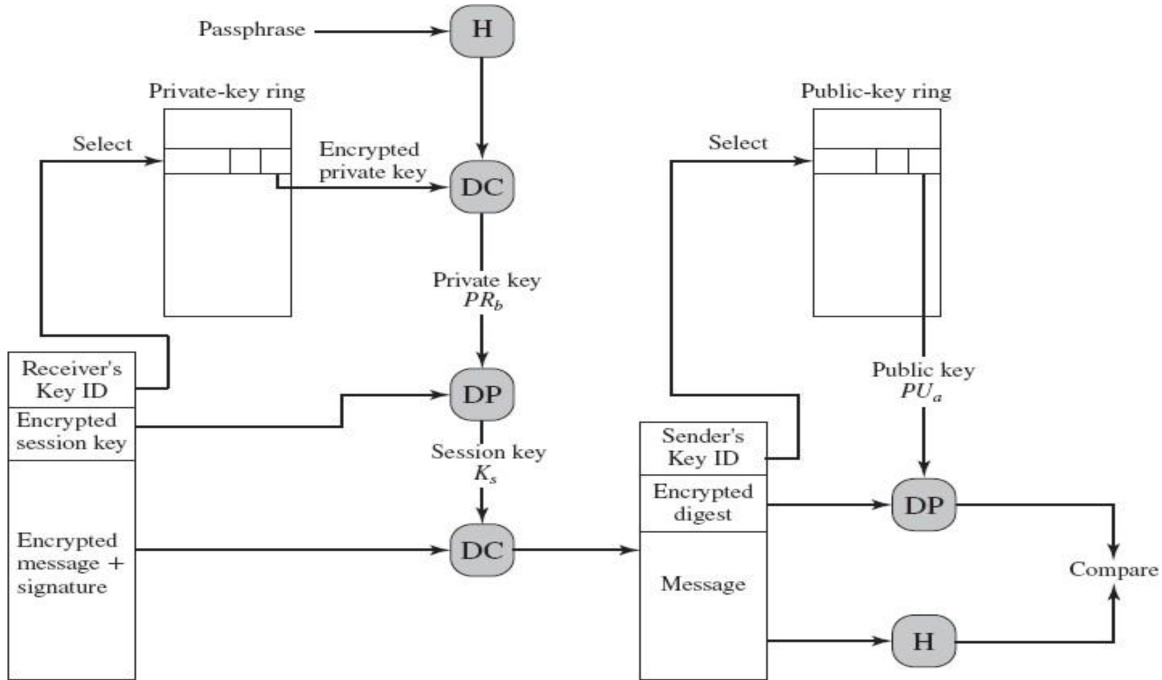


Fig5.6:PGPmessagereception

5.1.2. S/MIME

S/MIME (Secure/Multipurpose Internet Mail Extension) is a security enhancement to theMIMEInternete-mailformatstandard,based ontechnologyfrom RSADataSecurity.

5.1.2.1 MultipurposeInternetMailExtensions

MIME is an extension to the RFC 822 framework that is intended to address some of theproblems and limitations of the use of SMTP (Simple Mail Transfer Protocol) or some other mailtransfer protocoland RFC822forelectronicmail.

Following arethe limitationsof SMTP/822scheme:

1. SMTPcannottransmit executable filesorotherbinaryobjects.
2. SMTP cannot transmit text data that includes national language characters becausethese are represented by 8-bit codes with values of 128 decimal or higher, and SMTP is limitedto7-bitASCII.
3. SMTP serversmayrejectmailmessage overacertainsize.
4. SMTPgatewaysthattranslatebetweenASCIIandthecharactercodeEBCDICdonotuseaconsistentsetofmappings,resultingin translationproblems.
5. SMTPgatewaystoX.400electronicmailnetworkscannothandlenontextualdataincludedin X.400messages.
6. SomeSMTPimplementationsdonotadherecompletelytotheSMTPstandardsdefinedinRFC821.Common problemsinclude:

- Deletion, addition,orreorderingofcarriagereturnandlinefeed
- Truncatingorwrapping lines longerthan76 characters
- Removal oftrailingwhite space (tabandspacecharacters)
- Paddingoflinesinamessagetothesamelength

- Conversion oftabcharactersinto multiple spacecharacters

MIMEisintendedtoresolvetheseproblemsinamannerthatiscompatiblewithexistingRFC822implementations.ThespecificationisprovidedinRFCs2045through2049.

5.1.3 OVERVIEW

TheMIMEspecification includesthefollowingelements:

1. **Fivenewmessageheader**fieldsaredefined,whichmaybeincludedinanRFC822header.These fieldsprovideinformationaboutthebodyofthemessage.
2. **Anumberofcontentformats**aredefined,thusstandardizingrepresentationsthatsupportmultimediaelectronicmail.
3. **Transferencodings**aredefinedthatenabletheconversionofanycontentformatintoaformthatis protected fromalteration by the mailsystem.

Inthissubsection,weintroducethefivemessageheaderfields.Thenexttwosubsectionsdealwithcontent formatsand transferencodings.

Thefive headerfields definedin MIMEareas follows:

- **MIME-Version**:Musthavetheparametervalue1.0.Thisfieldindicatesthatthemessageconformsto RFCs2045 and2046.
- **Content-Type**:Describes thedatacontained inthe bodywith sufficientdetail.
- **Content-Transfer-Encoding**:Indicatesthetypeoftransformationthathasbeenusedtorepresentthebody ofthe messageina waythatisacceptable formailtransport.
- **Content-ID:**Usedto identifyMIMEentitiesuniquelyinmultiplecontexts.
- **Content-Description**:Atextdescriptionoftheobjectwiththebody;thisisusefulwhentheobjectisnotreadable(e.g.,audiodata).

5.1.3.1MIMEContentTypes

There are sevendifferent majortypesofcontentand a totalof15subtypes

MIMEContentTypes		
Type	**Subtype**	**Description**
Text	Plain	Unformattedtext; maybe ASCIIorISO8859.
	Enriched	Providesgreaterformatflexibility.
Multipart	Mixed	Thedifferentpartsare independentbutare tobetransmittedtogether.They shouldbepresentedtothereceiverintheorder thatthey appearinthemailmessage.
	Parallel	Differs from Mixed only in that no order is defined fordeliveringtheparts tothereceiver.
	Alternative	Thedifferentparts are alternativeversionsofthesameinformation. They are ordered in increasing faithfulness tothe original, and the recipient's mail system should displaythe "best"version tothe user.
	Digest	Similar to Mixed, but the default type/subtype of each partismessage/rfc822.
Message	rfc822	Thebodyisitselfanencapsulated messagethatconformstoRFC822.

	Partial	Used to allow fragmentation of large mail items, in a waythatistransparentto therecipient.
	External-body	Contains apointer toanobjectthat existselsewhere.
Image	jpeg	TheimageisinJPEGformat,JFIF encoding.
	Gif	TheimageisinGIFformat.
Video	mpeg	MPEGformat.
Audio	Basic	Single-channel 8-bit ISDN mu-law encoding at a samplerate of8kHz.
Application	PostScript	AdobePostscript.
	octet-stream	Generalbinarydataconsistingof8-bit bytes.

For the **text type** of body, no special software is required to get the full meaning of thetext,asidefromsupportoftheindicatedcharacterset.Theprimarysubtypeisplaintext,whichis simply a string of ASCII characters or ISO 8859 characters. The enriched subtype allowsgreaterformatting flexibility.

The **multipart type** indicates that the body contains multiple, independent parts. TheContent-Type header field includes a parameter, called boundary,that defines the delimiterbetweenbody parts.

The **multipart/digest subtype** is used when each of the body parts is interpreted as anRFC 822 message with headers. This subtype enables the construction of a message whoseparts are individual messages. For example, the moderator of a group might collect e-mailmessages from participants, bundle these messages, and send them out in one encapsulatingMIMEmessage.

The**messagetype**providesanumberofimportantcapabilitiesinMIME.Themessage/rfc822 subtype indicates that the body is an entire message, including header andbody. Despite the name of this subtype, the encapsulated message may be not only a simpleRFC822message,butalsoanyMIMEmessage.

The **message/partial subtype** enables fragmentation of a large message into a numberof parts, which must be reassembled at the destination. For this subtype, three parameters arespecified in the Content-Type: Message/Partial field: an id common to all fragments of the samemessage,asequence numberunique toeachfragment,andthe totalnumberoffragments.

The **message/external-body subtype** indicates that the actual data to be conveyed inthismessagearenotcontainedinthebody.Instead,thebodycontainstheinformationneededtoacces s thedata.

5.1.3.2. MIMETransferEncodings

MIMETransferEncodings

7bit	Thedataare all representedbyshort linesofASCII characters.8bit
	Thelinesareshort,buttheremaybenon-ASCII characters (octetswith thehigh-orderbit set).
binary	Notonlymaynon-ASCII characters bepresent butthelinesarenotnecessarilyshortenoughforSMTP transport.

quoted-printable	Encodesthedatainsuchawaythatifthedatabeingencodedare mostly ASCIItext,theencoded formofthedataremainslargely recognizablebyhumans.
base64	Encodesdatabymapping6-bit blocksofinputto8-bitblocks ofoutput,allofwhichareprintableASCIIcharacters.
x-token	Anamed nonstandardencoding.

The quoted-printable transfer encoding is useful when the data consists largely of octetsthat correspond to printable ASCII characters. In essence, it represents non safe characters bythe hexadecimal representation of their code and introduces reversible (soft) line breaks to limitmessagelinesto 76characters.

The base64 transfer encoding, also known as radix-64 encoding, is a common one forencoding arbitrary binary data in such a way as to be invulnerable to the processing by mailtransportprograms.

5.1.3.3. CanonicalForm

An important concept in MIME and S/MIME is that of canonical form. Canonical form is aformat, appropriate to the content type that is standardized for use between systems. This is incontrasttonative form,whichisa formatthatmay bepeculiartoaparticularsystem.

5.1.3.4. S/MIMEFunctionality

In terms of general functionality, S/MIME is very similar to PGP. Both offer the ability tosignand/orencryptmessages.

Functions:

S/MIMEprovides thefollowingfunctions:

- **Enveloped data:** This consists of encrypted content of any type and encrypted-contentencryptionkeysforone or morerecipients.
- **Signed data**: A digital signature is formed by taking the message digest of thecontent to be signed and then encrypting that with the private key of the signer.The content plus signature are then encoded using base 64 encoding. A signeddatamessagecanonly beviewedbyarecipientwithS/MIME capability.
- **Clear-signed data:** As with signed data, a digital signature of the content isformed.However,inthiscase,onlythedigitalsignatureisencodedusingbase 64.Asaresult,recipientswithoutS/MIMEcapabilitycanviewthemessagecontent,although theycannotverifythe signature.
- **Signed and enveloped data:** Encrypted data may be signed and signed data orclear-signeddatamay beencrypted.

5.1.3.5. CryptographicAlgorithms

Table 1 summarizes the cryptographic algorithms used in S/MIME. S/MIME uses thefollowingterminology,takenfrom RFC2119to specifytherequirementlevel:

- **Must:**Thedefinitionisanabsoluterequirementofthespecification.Animplementation must include this feature or function to be in conformance with thespecification.

- **Should:** There may exist valid reasons in particular circumstances to ignore thisfeature or function, but it is recommended that an implementation include the featureorfunction.

Table1:CryptographicAlgorithmsUsedinS/MIME	
Function	**Requirement**
Createamessagedigestto beusedinformingadigitalsignature.Encry ptmessagedigesttoformdigital signature.	MUSTsupport SHA-1. Receiver SHOULD support MD5 forbackwardcompatibility. SendingandreceivingagentsMUSTsupport DSS. SendingagentsSHOULDsupportRSAencry ption. ReceivingagentsSHOULDsupportverificat ionofRSAsignatureswithkeysizes512bitsto 1024 bits.
Encrypt session key for transmission withmessage.	Sending and receiving agents SHOULDsupportDiffie-Hellman. SendingandreceivingagentsMUSTsupport RSAencryptionwithkeysizes512bitsto 1024bits.
Encrypt message for transmission withone-time sessionkey.	SendingandreceivingagentsMUSTsupport encryptionwithtripleDES Sending agents SHOULD supportencryptionwithAES. Sending agents SHOULD supportencryptionwithRC2/40.
Create amessageauthenticationcode	ReceivingagentsMUSTsupportHMACwith SHA-1. Receiving agents SHOULD supportHMACwithSHA-1.

5.1.3.6S/MIMEMESSAGES

S/MIMEmakesuseofanumberofnewMIMEcontenttypes,whichareshowninTable 2. All of the new application types use the designation PKCS. This refers to a set of public-keycryptography specifications issued by RSA Laboratories and made available for the S/MIMEeffort.

Table2:S/MIMEContentTypes			
Type	**Subtype**	**SMIME Parameter**	**Description**
Multipart	Signed		Aclear-signedmessageintwoparts: one is the message and theotheristhe signature.
Application	PKCS 7-MIME	SignedData	AsignedS/MIMEentity.
	PKCS 7-MIME	Enveloped Data	Anencrypted S/MIMEentity.
	PKCS 7-MIME	degenerate signedData	Anentitycontainingonlypublic-keycertificates.
	PKCS 7-MIME	Compressed Data	AcompressedS/MIMEentity
	PKCS7-SIGNATURE	signed Data	The content type of the signaturesubpartofamultipart/signedmessage.

5.1.4 SECURINGAMIMEENTITY

S/MIME secures a MIME entity with a signature, encryption, or both. A MIME entity maybeanentiremessage(exceptfortheRFC822headers),oriftheMIMEcontenttypeismultipart, then a MIME entity is one or more of the subparts of the message. In all cases, themessageto besentisconvertedto canonical form.

In particular, for a given type and subtype, the appropriate canonical form is used for themessage content. For a multipart message, the appropriate canonical form is used for eachsubpart.

Theuseof transferencodingrequiresspecial attention.

1) EnvelopedData

1. Generate a pseudorandom session key for a particular symmetric encryption algorithm(RC2/40ortripleDES).
2. Foreachrecipient,encrypt thesessionkeywith the recipient'spublicRSAkey.
3. For each recipient, prepare a block known as Recipient Info that contains an identifier ofthe recipient's public-key certificate, an identifier of the algorithm used to encrypt thesessionkey,and the encryptedsession key.
4. Encryptthemessagecontentwith thesessionkey.

The Recipient Info blocks followed by the encrypted content constitute the envelopedData. This information is then encoded into base 64. To recover the encrypted message, therecipient first strips off the base64 encoding. Then the recipient's private key is used torecoverthe sessionkey. Finally, themessage content isdecrypted withthe session key.

2) SignedData

Thestepsfor preparingasignedData MIMEentityareasfollows:

- Selecta messagedigestalgorithm(SHAor MD5).

- Computethemessagedigest,or hashfunction,ofthecontenttobesigned
- Encryptthemessage digestwiththesigner'sprivatekey.
-
- 4.PrepareablockknownasSignerInfothatcontainsthesigner'spublic-keycertificate, an identifier of the message digest algorithm, an identifier of the algorithmusedto encryptthe messagedigest,andthe encryptedmessagedigest

To recover the signed message and verify the signature, the recipient first strips off thebase64encoding.Thenthesigner'spublic keyisusedto decryptthe messagedigest.

Therecipientindependentlycomputesthemessagedigestandcomparesittothedecrypted messagedigesttoverifythe signature.

3) ClearSigning

- Clearsigningisachieved using themultipartcontenttype withasignedsubtype.
- Aswasmentioned,thissigningprocessdoesnotinvolvetransformingthemessagetobesigned, sothatthemessage issent"intheclear."
- Thus, recipients with MIME capability but not S/MIME capability are able to read theincomingmessage.

Amultipart/signedmessagehastwo parts.

The first partcanbeanyMIMEtypebutmust beprepared sothatit will notbealtered during transfer from source to destination. This means that if the first part is not 7bit,thenitneedsto beencodedusingbase64 orquoted-printable.

This second part has a MIME content type of application and a subtype of PKCS7-signatureThe protocol parameter indicates that this is a two-part clear-signed entity. Thereceivercanverifythesignaturebytakingthemessagedigestofthefirstpartandcomparingthisto themessage digestrecoveredfromthe signatureinthe second part.

5.1.4.1 RegistrationRequest

The user will apply to a certification authority for a public-key certificate. The S/MIMEentity isusedto transferacertificationrequest.

- The certification request includes certification Request Info block, followed by anidentifier of the public-key encryption algorithm, followed by the signature of thecertificationRequestInfo block,made usingthesender's privatekey.
- The certification Request Info block includes a name of the certificate subject (theentity whose public key is to be certified) and a bit-string representation of the user'spublickey.

Certificates-OnlyMessage

Amessagecontainingonlycertificatesoracertificaterevocationlist(CRL)canbesentin response to a registration request. The message is an application/PKCS7-MIME type/subtypewith an SMIME-type parameter of degenerate. The steps involved are the same as those forcreating a signed Data message, except that there is no message content and the signer Infofieldis empty.

S/MIMECertificateProcessing

S/MIMEusespublic-keycertificatesthatconformtoversion3ofX.509Thekey-managementschemeusedbyS/MIMEisinsomewaysahybridbetweenastrictX.509certificationhierarchy andPGP'sweboftrust.

UserAgentRole

AnS/MIMEuserhas**several key-managementfunctions**toperform:

Keygeneration:

Theuserofsomerelatedadministrativeutility(e.g.,oneassociatedwithLANmanagement)MUSTbecapableofgeneratingakeypairfromagoodsourceofnondeterministic randominputand beprotectedinasecure fashion.

Registration:

Auser'spublickeymustberegisteredwithacertificationauthorityinordertoreceivean X.509public-keycertificate.

Certificatestorageandretrieval:

A user requires access to a local list of certificates in order to verify incoming signaturesand to encrypt outgoing messages. Such a list could be maintained by the user or by some localadministrativeentityonbehalfofanumberofusers.

VeriSignCertificates

There are several companies that provide certification authority (CA) services. There area numberofInternet-basedCAs,includingVeriSign,GTE, and theU.S.PostalService.

VeriSign provides a CA service that is intended to be compatible with S/MIME and avariety of other applications. VeriSign issues X.509 certificates with the product name VeriSignDigitalID.

TheinformationcontainedinaDigitalIDdependsonthetypeofDigitalIDanditsuse.Ata minimum,each DigitalIDcontains

- Owner'spublic key
- Owner's nameoralias
- Expirationdate of the DigitalID
- Serial number oftheDigital ID
- Name of the certificationauthoritythatissuedtheDigitalID
- Digitalsignatureofthecertification authoritythat issuedtheDigitalID

DigitalIDscanalsocontainotheruser-suppliedinformation,including

- Address
- E-mailaddress
- Basic registrationinformation (country,zip code,age,andgender)

VeriSign provides three levels, or classes, of security for public-key certificates. A user requestsa certificate online at VeriSign's Web site or other participating Web sites. Class 1 and Class 2requests are processed on line, and in most cases take only a few seconds to approve. Briefly,thefollowingproceduresareused:

- ForClass1DigitalIDs,VeriSignconfirmstheuser'se-mailaddressbysendingaPINand DigitalIDpick-upinformationtothee-mail addressprovided intheapplication.

- For Class 2 Digital IDs, VeriSign verifies the information in the application through anautomated comparison with a consumer database in addition to performing all of thecheckingassociatedwithaClass1DigitalID.

    - Finally,confirmationissenttothespecifiedpostaladdressalertingtheuserthataDigital IDhasbeenissuedinhisorhername.

- ForClass3DigitalIDs,VeriSignrequiresahigherlevelofidentityassurance.Anindividual must prove his or her identity by providing notarized credentials or applying inperson.

5.1.4.2 EnhancedSecurityServices

Threeenhancedsecurityservices have been proposedin anInternetdraft.

Signedreceipts:

A signed receipt may be requested in a Signed Data object. Returning a signed receiptprovidesproofofdeliverytotheoriginatorofamessageandallowstheoriginatortodemonstrate to a thirdpartythattherecipientreceivedthemessage.

Securitylabels:

AsecuritylabelmaybeincludedintheauthenticatedattributesofaSignedDataobject.A security label is a set of security information regarding the sensitivity of the content that isprotected by S/MIME encapsulation. The labels may be used for access control, by indicatingwhichusersarepermittedaccessto anobject.

Securemailinglists:

When a user sends a message to multiple recipients, a certain amount of per-recipientprocessingis required,includingtheuse ofeachrecipient'spublic key.

The user can be relieved of this work by employing the services of an S/MIME Mail ListAgent(MLA).AnMLAcantake asingleincomingmessage,performthe recipient-specificencryption foreach recipient,andforwardthe message.

The originator of a message need only send the message to the MLA, with encryptionperformedusingtheMLA'spublickey.

5.1.5 NONREPUDIATION

Non-repudiation is the assurance that someone cannot deny something. Typically, non-repudiation refers to the ability to ensure that a party to a contract or a communication cannotdeny the authenticity of their signature on a document or the sending of a message that theyoriginated.

To repudiate means to deny. On the Internet, a digital signature is used not only toensure that a message or document has been electronically signed by the person that purportedto sign the document, but also, since a digital signature can only be created by one person, toensure thataperson cannotlaterdenythattheyfurnishedthe signature.

Since no security technology is absolutely fool-proof, some experts warn that a digitalsignaturealonemaynotalwaysguaranteenon-repudiation.Itissuggestedthatmultipleapproaches be used, such as capturing unique biometric information and other data about thesenderorsignerthatcollectively wouldbedifficulttorepudiate.

Emailnon-repudiationinvolvesmethodssuchasemailtrackingthatisdesignedtoensurethatthesendercannotdenyhavingsentamessageand/orthattherecipientcannotdenyhavingreceivedit

5.2IPSECURITY

5.2.1 OVERVIEWOFIPSEC

5.2.1.1 ApplicationsofIPSec

IPSecprovidesthecapabilitytosecurecommunicationsacrossaLAN,acrossprivateandpublic WANs,andacross theInternet.Examplesofitsuseinclude the following:

- Securebranchofficeconnectivityoverthe Internet
- 2Secure remoteaccess overthe Internet
- Establishing extranetandintranetconnectivitywithpartners
- Enhancingelectroniccommercesecurity

5.2.1.2 BenefitsofIPSec:

- WhenIPSec isimplementedinafirewallor router,itprovides strong security
- IPSec in a firewall is resistant to bypass if all traffic from the outside must use IP,andthefirewallistheonlymeansofentrancefromtheInternetintotheorganization.
- IPSecisbelowthetransportlayer(TCP,UDP)andsoistransparenttoapplications. There is no need to change software on a user or server systemwhenIPSecisimplementedinthefirewallorrouter.
- IPSeccanbetransparenttoendusers. Thereisnoneedtotrainusersonsecurity mechanisms
- IPSeccanprovidesecurityforindividual users ifneeded.

5.2.1.3 RoutingApplications

IPSec can play a vital role in the routing architecture required for internet

working.Thefollowingareexamplesofthe useofIPSec.IPSeccanassure that

- A router advertisement (a new router advertises its presence) comes from anauthorizedrouter
- A neighbor advertisement (a router seeks to establish or maintain a neighborrelationship with a router in another routing domain) comes from an authorizedrouter.
- A redirectmessagecomes fromtherouter towhich theinitial packetwas sent.
- A routing updateisnot forged.

5.2.2 IPSECURITYARCHITECTURE

5.2.2.1 IPSecDocuments

TheIPSecspecificationconsistsofnumerousdocuments.Themostimportantofthese,issuedin Novemberof1998,areRFCs2401,2402,2406,and 2408:

- **RFC2401:**Anoverviewofasecurity architecture
- **RFC2402:** DescriptionofapacketauthenticationextensiontoIPv4and IPv6
- **RFC2406:**Description ofapacketencryption extensiontoIPv4 andIPv6
- **RFC2408:**SpecificationofkeymanagementcapabilitiesTh

edocumentsaredividedintoseven groups:

5.2.2.2 Architecture

Coversthegeneralconcepts,securityrequirements,definitions,andmechanismsdefiningIPSectechnology.

EncapsulatingSecurityPayload(ESP):

Covers the packet format and general issues related to the use of the ESP for packetencryptionand,optionally,authentication.

AuthenticationHeader(AH):

CoversthepacketformatandgeneralissuesrelatedtotheuseofAHforpacketauthentication.

EncryptionAlgorithm:

A setofdocuments that describehowvariousencryption algorithmsare usedforESP.

AuthenticationAlgorithm:

A set of documents that describe how various authentication algorithms are used for AHand fortheauthenticationoptionofESP.

KeyManagement:

Documents thatdescribekeymanagementschemes.

DomainofInterpretation(DOI):

Contains values needed for the other documents to relate to each other. These includeidentifiersforapprovedencryptionandauthenticationalgorithms,aswellasoperationalparameters such askey lifetime.

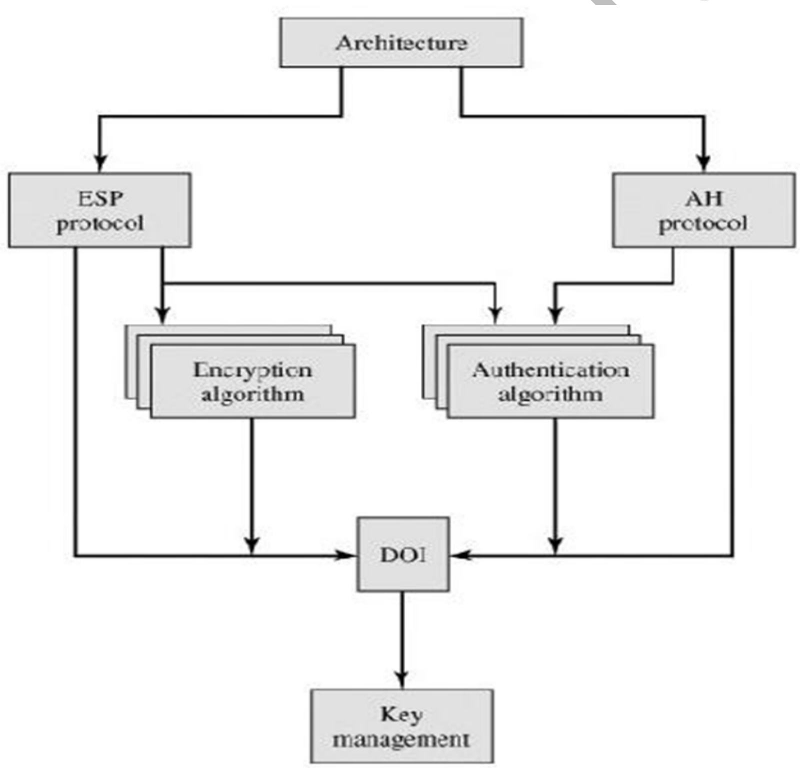


Fig5.7:IPsecurityDocumentoverview

IPSecServices

IPSec provides security services at the IP layer by enabling a system to select requiredsecurity protocols, determine the algorithm(s) to use for the service(s), and put in place anycryptographic keysrequired toprovidethe requestedservices.

Two protocolsareusedto providesecurity:

- Anauthenticationprotocol:Designatedbytheheaderoftheprotocol,AuthenticationHeader(AH);
- Encryption/authenticationprotocoldesignatedbytheformatofthepacketforthatprotocol,EncapsulatingSecurityPayload(ESP).

Theservicesare

- Accesscontrol
- Connectionlessintegrity
- Dataoriginauthentication
- Rejectionof replayed packets(aformofpartialsequenceintegrity)
- Confidentiality(encryption)
- Limitedtrafficflowconfidentiality

SecurityAssociations

A key concept that appears in both the authentication and confidentiality mechanisms forIP is the security association (SA). An association is a one-way relationship between a senderandareceiverthataffordssecurity servicestothe traffic carriedonit.

Asecurityassociation isuniquelyidentifiedbythreeparameters:

- SecurityParametersIndex(SPI)
- IPDestination Address
- SecurityProtocolIdentifier

SAParameters

Asecurityassociationisnormallydefined bythefollowingparameters:

a) SequenceNumberCounter:

A32-bitvalueusedtogenerate theSequenceNumber fieldinAHorESPheaders.

b) SequenceCounterOverflow:

A flag indicating whether overflow of the Sequence Number Counter should generate anauditableeventandpreventfurthertransmissionofpackets on thisSA.

c) Anti-ReplayWindow:

UsedtodeterminewhetheraninboundAH orESPpacket isareplay.

d) AHInformation:

Authentication algorithm, keys, key lifetimes, and related parameters being used with AH(requiredforAHimplementations).

e) ESPInformation:

Encryption and authentication algorithm, keys, initialization values, key lifetimes, andrelatedparametersbeingusedwithESP(requiredforESPimplementations).

f) LifetimeofThisSecurityAssociation:

A time interval or byte count after which an SA must be replaced with a new SAorterminated,plusan indicationofwhichoftheseactionsshouldoccur .

g) IPSecProtocolMode: Tunnel, transport.
h) PathMTU: Anyobserved pathmaximumtransmissionunitandagingvariables.

5.2.2.3. ModesofTransfer

BothAHandESPsupporttwomodes ofuse:transportand tunnelmode.

TransportMode:

Transport mode provides protection primarily for upper-layer protocols. That is, transportmodeprotectionextends to the payloadofan IP packet.

Tunnel Mode:

Tunnel mode provides protection to the entire IP packet. To achieve this, after the AH orESP fields are added to the IP packet, the entire packet plus security fields is treated as thepayloadofnew "outer"IPpacketwithanewouterIPheader.

The entire original, or inner, packet travels through a "tunnel" from one point of an IPnetwork to another; no routers along the way are able to examine the inner IP header. Becausethe original packet is encapsulated, the new, larger packet may have totally different source anddestinationaddresses,addingtothe security.

5.2.2.4. AuthenticationHeader

TheAuthenticationHeaderprovidessupportfordataintegrityandauthenticationofIPpackets.The AuthenticationHeaderconsistsofthe followingfields:

- **NextHeader(8bits):**Identifiesthetype ofheaderimmediatelyfollowingthis header.
- **PayloadLength(8bits):**LengthofAuthenticationHeaderin32-bit words, minus2.
- **Reserved(16bits):**Forfuture use.
- **SecurityParametersIndex (32bits):**Identifiesasecurity association.
- **SequenceNumber(32bits):**Amonotonically increasingcounter value.
- **AuthenticationData(variable):**Avariable-lengthfield(mustbeanintegralnumberof32-bitwords)thatcontainstheIntegrity Check Value(ICV),orMAC

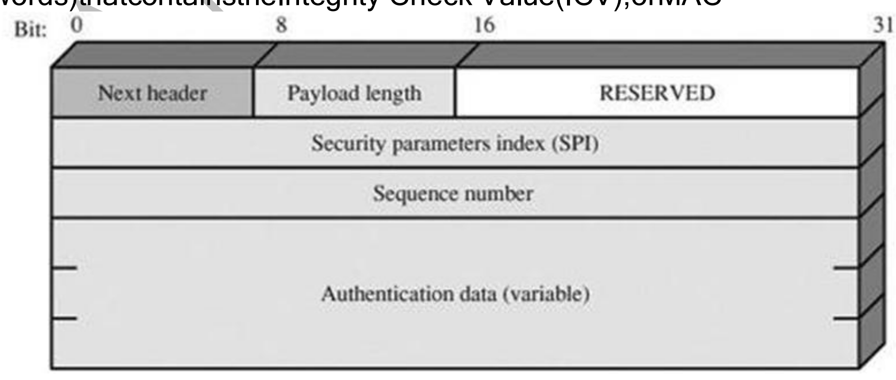


Fig5.8:IPSecAuthenticationHeader

Anti-ReplayService

Areplayattackisoneinwhichanattackerobtainsacopyofanauthenticatedpacketandlatertransmitsitto theintendeddestination.

When a new SA is established, the sender initializes a sequence number counter to 0.Each time that a packet is sent on this SA,the sender increments the counter and places thevalueintheSequenceNumberfield.Thus,thefirstvalueto beusedis1.

If anti-replay is enabled (the default), the sender must not allow the sequence number tocycle past 2^{32}-1 back to zero. Otherwise, there would be multiple valid packets with the samesequence number. If the limit of 2^{32-}1 is reached, the sender should terminate this SA andnegotiatea new SAwithanew key.

IntegrityCheck Value

The Authentication Data field holds a value referred to as the Integrity Check Value. TheICV is a message authentication code or a truncated version of a code produced by a MACalgorithm.

Transport andTunnelModes

For transport mode AH using IPv4, the AH is inserted after the original IP header andbefore the IP payload

FortunnelmodeAH,theentireoriginalIPpacketisauthenticated,andtheAHisinsertedbetweent heoriginalIP headeranda new outerIP header

5.2.2.5. EncapsulatingSecurityPayload

TheEncapsulatingSecurityPayloadprovidesconfidentialityservices,includingconfidentiality ofmessagecontents andlimitedtrafficflowconfidentiality.

Thediagram showstheformatofanESPpacket.Itcontainsthefollowingfields:

- **SecurityParametersIndex (32bits):**Identifiesasecurity association.
- **Sequence Number (32 bits):** A monotonically increasing counter value; this provides ananti-replayfunction,asdiscussedforAH.
- **PayloadData(variable):**Thisisatransport-levelsegment(transportmode)orIPpacket(tunnel mode)thatisprotected byencryption.
- **Padding(0255bytes):**Thepurposeofthisfield is discussedlater.
- **Pad Length (8 bits):** Indicates the number of pad bytes immediately precedingthisfield.
- **Next Header (8 bits):** Identifies the type of data contained in the payload datafield by identifying the first header in that payload (for example, an extensionheaderinIPv6,oranupper-layerprotocolsuchas TCP).
- **Authentication Data (variable):** A variable-length field (must be an integral number of32-bit words) that contains the Integrity Check Value computed over the ESP packetminus theAuthentication Datafield.

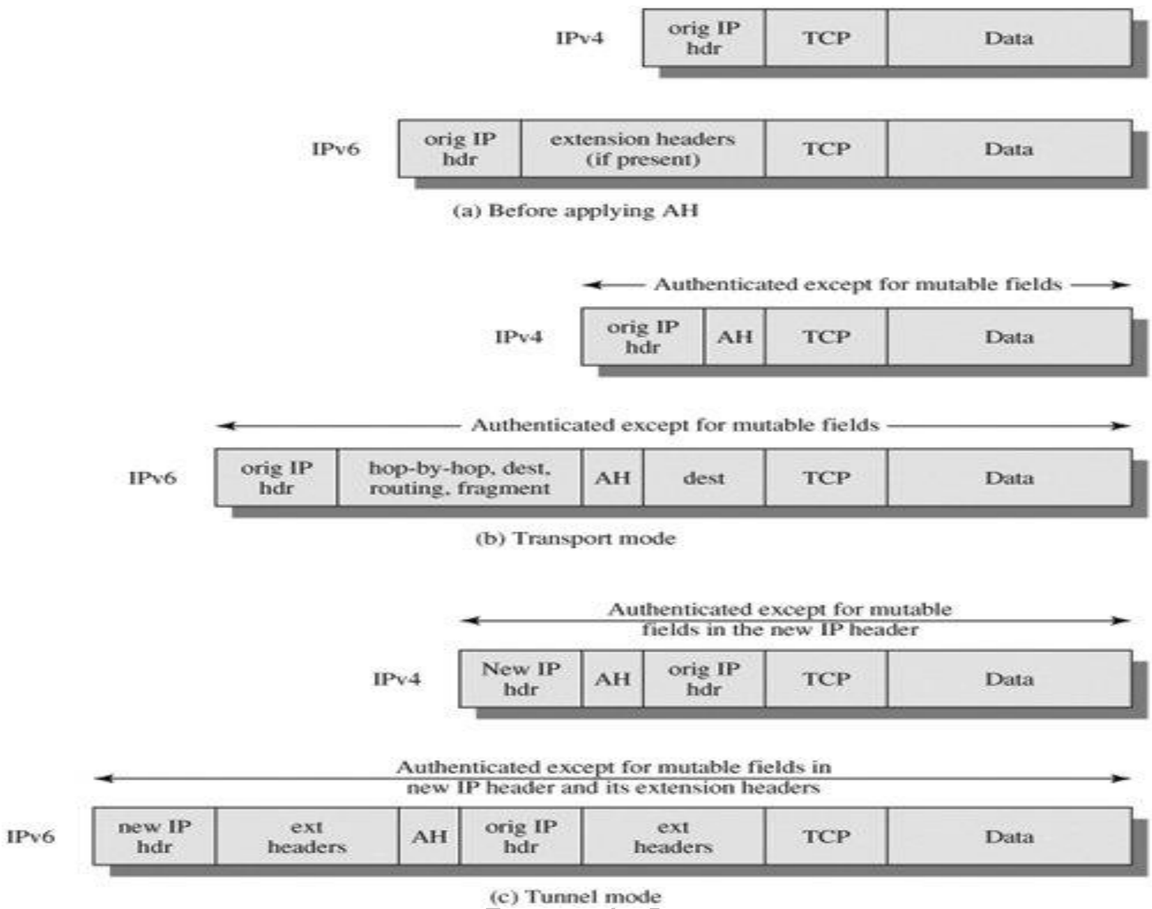


Fig5.9:ScopeofAHAuthentication

Padding:

ThePaddingfieldserves severalpurposes:

- If an encryption algorithm requires the plaintext to be a multiple of some number ofbytes (e.g., the multiple of a single block for a block cipher), the Padding field is usedto expand the plaintext (consisting of the Payload Data, Padding, Pad Length, andNextHeaderfields)to therequired length.
- The ESP format requires that the Pad Length and Next Header fields be right alignedwithin a 32-bit word. Equivalently, the ciphertext must be an integer multiple of 32bits.The Paddingfieldis usedtoassurethisalignment.
- Additional paddingmay be added to provide partial traffic flow confidentiality byconcealingtheactuallengthofthepayload.

Transport andTunnelModes

ESPservicecanbeused.Intheupperpartofthefigure,encryption(andoptionallyauthentication)isprovideddirectlybetweentwohosts.

Thediagramshowshowtunnelmodeoperationcanbeusedtosetupavirtualprivatenetwork.

Inthisexample,anorganizationhasfourprivatenetworksinterconnectedacrosstheInternet.

HostsontheinternalnetworksusetheInternetfortransportofdatabutdonotinteractwithotherInternet-based hosts.

Byterminatingthetunnelsatthesecuritygatewaytoeachinternalnetwork,theconfigurationallowsthehoststoavoidimplementingthesecuritycapability.Theformertechniqueissupportbyatransportmode SA,whilethelattertechniqueusesa tunnelmode SA.**TransportModeESP:**

For thismodeusingIPv4, theESPheaderis inserted into the IPpacketimmediatelyprior to the transport-layer header (e.g., TCP, UDP, ICMP) and an ESP trailer (Padding, PadLength, and Next Header fields) is placed after the IP packet; if authentication is selected, theESPAuthentication Datafieldisadded afterthe ESPtrailer.

The entire transport-level segment plus the ESP trailer are encrypted. Authenticationcoversallofthe ciphertextplusthe ESPheader.

For thismodeusingIPv4, theESPheaderis inserted into the IPpacketimmediatelyprior to the transport-layer header (e.g., TCP, UDP, ICMP) and an ESP trailer (Padding, PadLength, and Next Header fields) is placed after the IP packet; if authentication is selected, theESPAuthentication Datafieldisadded afterthe ESPtrailer.

The entire transport-level segment plus the ESP trailer are encrypted. Authenticationcoversallofthe ciphertextplusthe ESPheader.

Transport mode operation provides confidentiality for any application that uses it, thusavoiding the need to implement confidentiality in every individual application. This mode ofoperation is also reasonably efficient, adding little to the total length of the IP packet. Onedrawbackto thismodeisthatitispossibleto dotrafficanalysisonthetransmittedpackets.

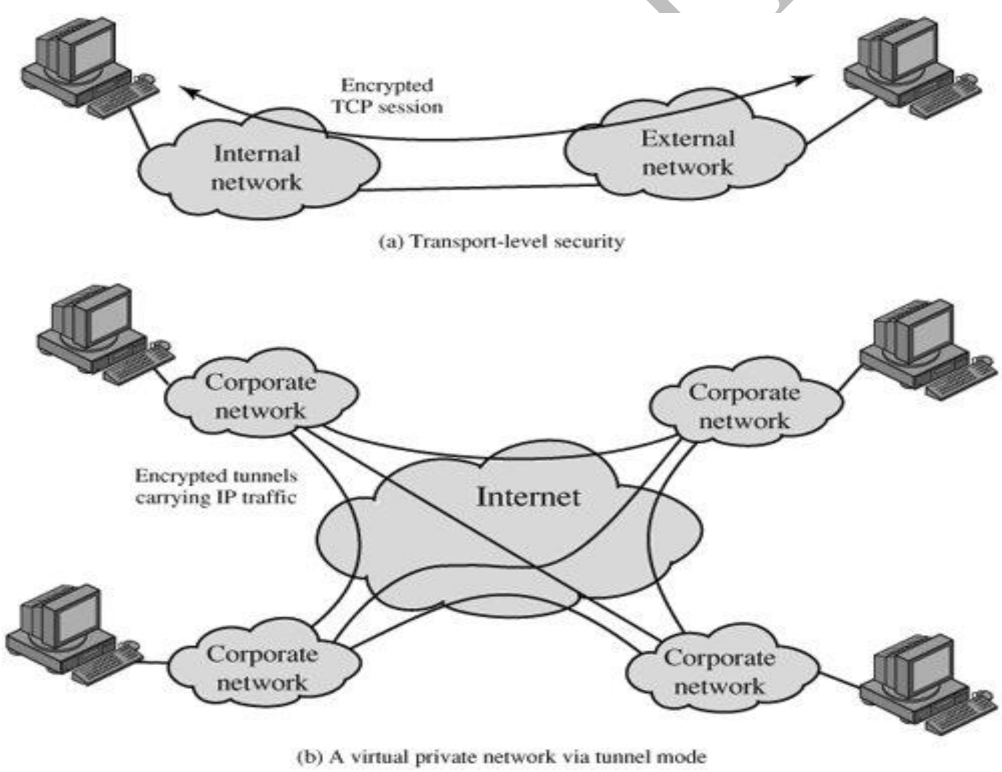


Fig5.10:TransportMode

Tunnel ModeESP

Tunnel mode ESP is used to encrypt an entire IP packet. For this mode, the ESP headeris prefixed to the packet and then the packet plus the ESP trailer is encrypted. This method canbeusedtocountertrafficanalysis.

Because the IP header contains the destination address and possibly source routingdirectives and hop-by-hop option information, it is not possible simply to transmit the encryptedIP packet prefixed by the ESP header. Intermediate routers would be unable to process such apacket.

Therefore, it is necessary to encapsulate the entire block (ESP header plus cipher textplus Authentication Data, if present) with a new IP header that will contain sufficient informationfor routingbutnot fortrafficanalysis.

5.2.3 COMBININGSECURITYASSOCIATIONS

AnindividualSAcanimplementeithertheAHorESPprotocolbutnotboth.Sometimesa particular traffic flow will call for the services provided by both AH and ESP.Further, aparticulartrafficflowmayrequireIPsecservicesbetweenhostsand,forthatsameflow,separateservic esbetweensecuritygateways,suchasfirewalls.

Securityassociationsmaybecombined intobundlesintwo ways:

Transportadjacency:

Referstoapplyingmorethanonesecurityprotocoltothesame IPpacketwithoutinvokingtunneli ng.ThisapproachtocombiningAHandESPallowsforonlyonelevelofcombination; further nesting yields no added benefit since the processing is performed at oneIPsecinstance:the (ultimate)destination.

Iteratedtunneling:

Refers to the application of multiple layers of security protocols effected through IPtunneling. This approach allows for multiple levels of nesting, since each tunnel can originate orterminateata differentIPsecsitealongthe path.

The two approaches can be combined, for example, by having a transport SA betweenhosts travelpartoftheway throughatunnel SAbetweensecuritygateways.

One interesting issue that arises when considering SA bundles is the order in whichauthentication and encryption may be applied betweena given pair ofendpointsand the waysof doing so. We examine that issue next. Then we look at combinations of SAs that involve atleastonetunnel.

AuthenticationplusConfidentiality

Encryption andauthenticationcanbecombinedinordertotransmit anIPpacket thathasbothconfidentiality and authenticationbetween hosts.Welookatseveral approaches.

5.2.3.1 ESPwithAuthenticationOption

This approach is illustrated in diagram. In this approach, the user first applies ESP to thedata to be protected and then appends the authentication data field. There are actually twosubcases:

TransportmodeESP: Authentication and encryption apply to the IP payload deliveredto thehost,butthe IPheaderisnotprotected.

Tunnel mode ESP: Authentication applies to the entire IP packet delivered to the outerIP destination address (e.g., a firewall), and authentication is performed at that destination. TheentireinnerIPpacketisprotectedbytheprivacymechanismfordeliverytotheinnerIPdestination.

Forboth cases, authentication appliestothecipher textratherthanthe plaintext.

Transport Adjacency

Anotherway toapplyauthentication afterencryptionistouse twobundledtransportSAs, with the inner being an ESP SA and the outer being an AH SA. In this case, ESP is usedwithoutits authentication option.Because the innerSA is a transportSA, encryption is appliedto the IP payload. The resulting packet consists of an IP header (and possibly IPv6 headerextensions)followedbyanESP.AHisthenappliedintransportmode,

sothatauthenticationcoverstheESPplus theoriginalIPheader(andextensions)except for mutable fields. The advantage of this approach over simply using a single ESP SAwith the ESP authentication option is that the authentication covers more fields, including thesource and destination IP addresses.The disadvantage is the overhead of two SAs versus oneSA.

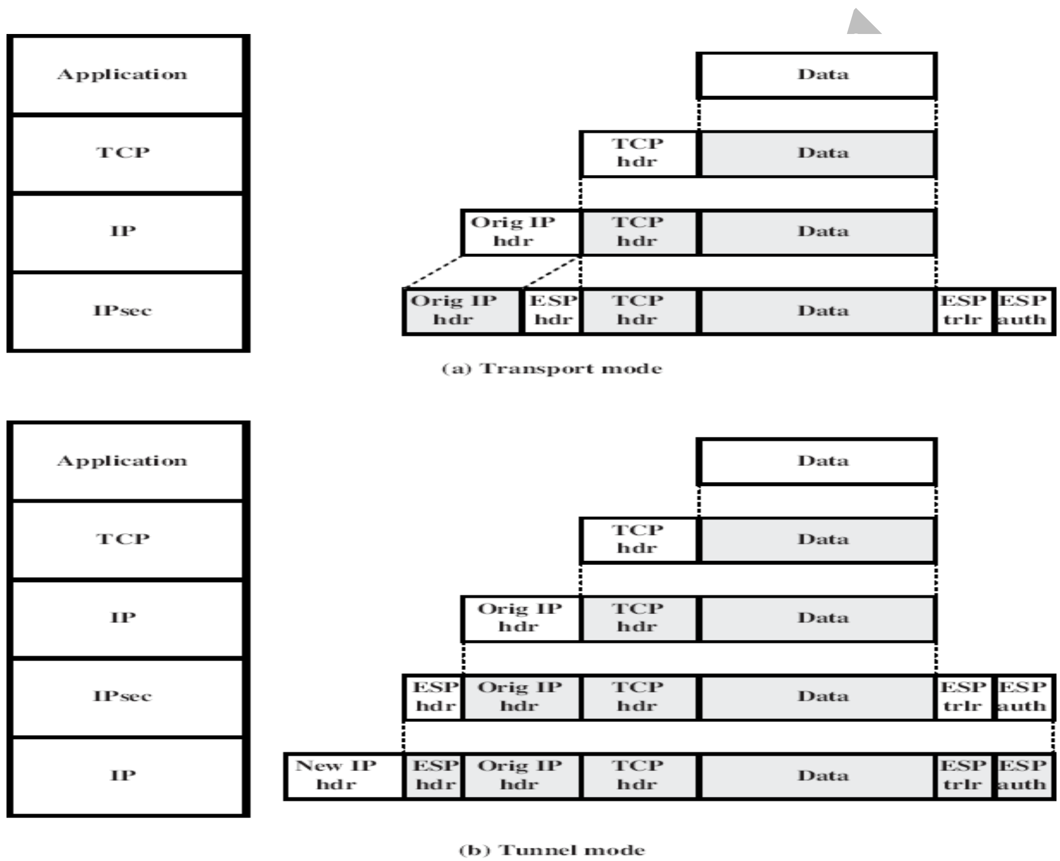


Fig 5.11: Protocol Operation for

ESPBasicCombinations ofSecurityAssociations

TheIPsecArchitecturedocumentlistsfourexamplesofcombinationsofSAsthatmustbe supported by compliant IPsec hosts (e.g., workstation, server) or security gateways (e.g.firewall,router).Theseareillustrated inFigure.

The lower part of each case in the figure represents the physical connectivity of theelements;theupperpartrepresentslogicalconnectivityviaoneormorenestedSAs.EachSA canbeeitherAHorESP.Forhost-to-hostSAs,themodemaybeeithertransportortunnel;otherwiseitmustbe tunnelmode.

Case1: All securityisprovidedbetween endsystems that implementIPsec.

ForanytwoendsystemstocommunicateviaanSA,theymustsharetheappropriatesecretkeys.Amongthepossiblecombinationsare

- AH intransportmode
- ESPintransportmode
- ESPfollowedby AHintransportmode(anESPSAinsideanAHSA)
- Anyoneofa,b,orcinsideanAHorESPintunnelmode

Wehavealreadydiscussedhowthesevariouscombinationscanbeusedtosupportauthentication,encryption, authenticationbeforeencryption, andauthenticationafterencryption.

Case2: Security isprovided only between gateways(routers, firewalls,etc.)and nohosts implement IPsec. This case illustrates simple virtual private network support. The securityarchitecture document specifies that only a single tunnel SA is needed for this case. The tunnelcould support AH, ESP, or ESP with the authentication option. Nested tunnels are not required,becausetheIPsecservicesapplyto theentireinnerpacket.

Case 3: This builds on case 2 by adding end-to-end security. The same combinationsdiscussed for cases 1 and 2 are allowed here. The gateway-to-gateway tunnel provides eitherauthentication, confidentiality, or both for all traffic between end systems.When the gateway-to-gateway tunnel is ESP, it also provides a limited form of traffic confidentiality. Individual hostscan implement any additional IPsec servicesrequired for given applications or given users bymeansofend-to-end SAs.

Case 4: This provides support for a remote host that uses the Internet to reach anorganization"s firewall and then to gain access to some server or workstation behind the firewall.Only tunnel mode is required between the remote host and the firewall. As in case 1, one or twoSAsmay beusedbetweentheremotehostand thelocalhost.

5.2.4 KEYMANAGEMENT

The key management portion of IPSec involves the determination and distribution ofsecret keys.Twotypesofkeymanagement:

Manual: A system administrator manually configures each system with its own keys andwiththekeysofothercommunicatingsystems.Thisispracticalforsmall,relativelystaticenvironments.

Automated: An automated system enables the on-demand creation of keys for SAs andfacilitatestheuse ofkeysinalarge distributedsystemwithanevolvingconfiguration.

ThedefaultautomatedkeymanagementprotocolforIPSecisreferredtoasISAKMP/Oakleyand consistsofthe followingelements:

Oakley Key Determination Protocol: Oakley is a key exchange protocol based on theDiffie-Hellman algorithm but providingadded security. Oakley is generic in thatit does notdictate specificformats.

Internet Security Association and Key Management Protocol (ISAKMP):ISAKMPprovides a framework for Internet key management and provides the specific protocol support,includingformats,for negotiationofsecurityattributes.

(firewall, router).These are illustrated in Figure .The lower part

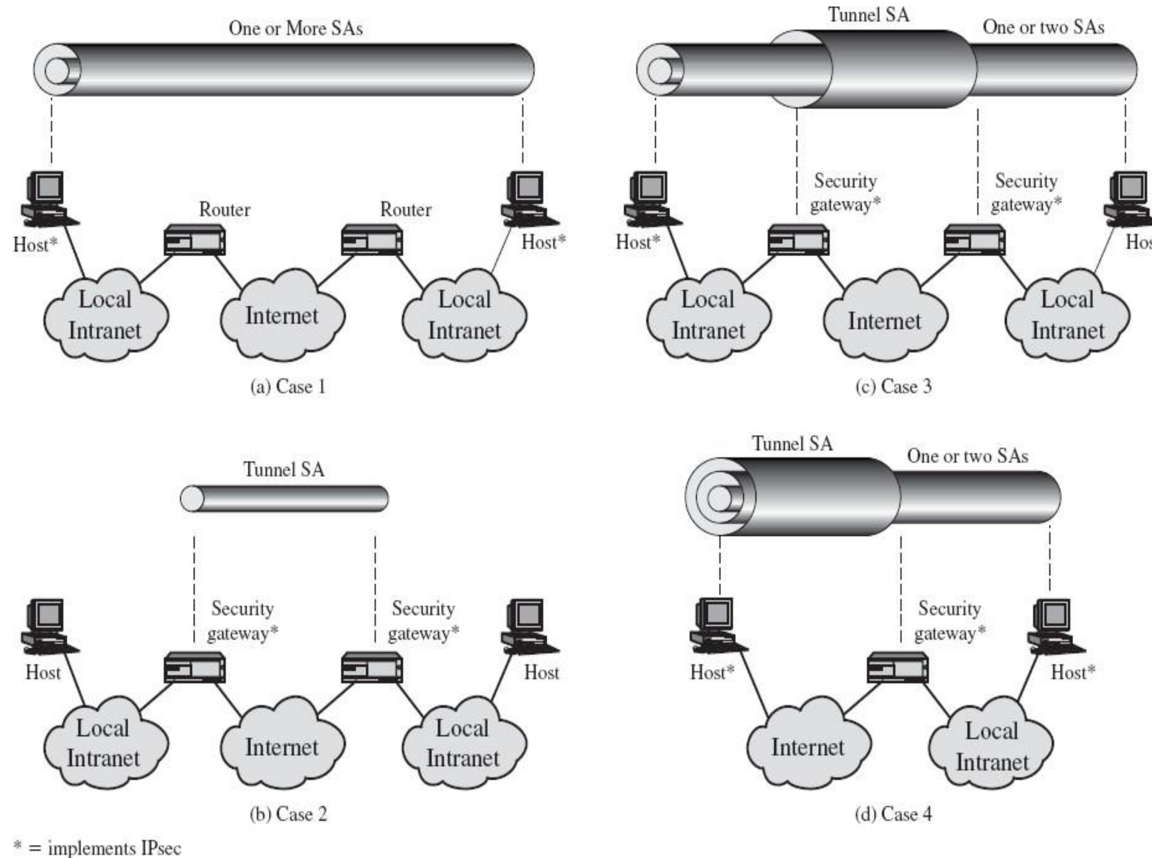


Fig5.12:BasicCombinationsofSecurityAssociations

5.2.4.1. OakleyKeyDeterminationProtocol

Oakley is a refinement of the Diffie-Hellman key exchange algorithm. There is prioragreement on two global parameters: q, a large prime number; anda primitive root of q. Aselectsarandominteger $X_A$asits privatekey,andtransmits toB its public key $Y_A{}^{XA}$modq.

Similarly,BselectsarandomintegerX$_B$asitsprivatekeyandtransmitstoAitspublickey$Y_B{}^{XB}$mod q.Eachsidecannow computethesecretsessionkey:

$K = (Y_B)^{XA} \text{mod} q = (Y_A)^{XB} \text{mod} q = {}^{XAXB} \text{mod} q$

TheDiffie-Hellmanalgorithmhastwoattractivefeatures:

- Secretkeysarecreatedonlywhenneeded.Thereisnoneedtostore secretkeysforalongperiodoftime,exposingthemtoincreasedvulnerability.
- Theexchangerequiresnopreexistinginfrastructureotherthananagreementontheglob alparameters.

5.2.4.2. FeaturesofOakley

TheOakleyalgorithmischaracterizedbyfive important features:

- It employsa mechanism knownascookiestothwart cloggingattacks.
- Itenablesthetwopartiestonegotiateagroup;this,inessence,specifiestheglobalparam etersoftheDiffie-Hellman key exchange.

- Itusesnonces toensureagainstreplayattacks.
- Itenablesthe exchangeofDiffie-Hellman publickeyvalues.
- It authenticates the Diffie-Hellman exchange to thwart man-in-the-middle

attacks.Threedifferentauthenticationmethods canbe usedwithOakley:

- Digitalsignatures
- Public-keyencryption
- Symmetric-keyencryption

5.2.4.3. ISAKMP

ISAKMPdefinesproceduresandpacketformatstoestablish,negotiate,modify,anddeletesecurityassociations.

ISAKMPHeaderFormat

An ISAKMP messageconsistsofanISAKMP header followedbyoneor morepayloads.Itconsistsofthe followingfields:

- **InitiatorCookie(64bits):**CookieofentitythatinitiatedSAestablishment,SAnotification,orSAdeletion.
- **ResponderCookie(64bits)**:Cookieofrespondingentity;nullinfirstmessagefrominitiator.
- **NextPayload(8bits):**Indicatesthetypeofthefirstpayloadinthemessage;payloadsare discussedinthenextsubsection.
- **MajorVersion(4bits)**: Indicatesmajorversion of ISAKMPin use.
- **MinorVersion(4bits):**Indicatesminorversion inuse.
- **ExchangeType(8bits):** Indicatesthetypeofexchange.
- **Flags(8bits):**Indicates specificoptionssetforthisISAKMPexchange.
- **MessageID(32bits):** Unique IDfor this message.

- **Length(32bits):**Lengthoftotal message(headerplus allpayloads)in octets.

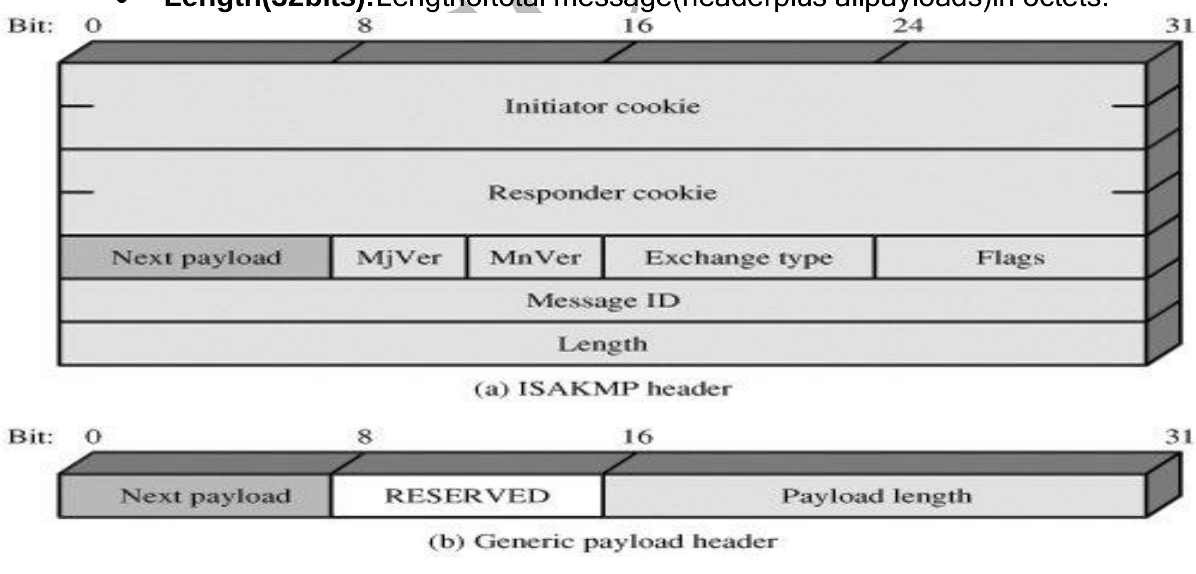


Fig5.13:ISAKMPHeaderFormat

5.3WEBSECURITY

5.3.1 WEBSECURITYCONSIDERATIONS

TheWorldWideWebisfundamentallyaclient/serverapplicationrunningovertheInternetandTCP/IPintranets.

WebSecurityThreats

AComparisonofThreatsontheWeb

	Threats	Consequences	Countermeasures
Integrity	ModificationofuserdataTrojanhorsebrowser Modification of memoryModification ofmessage traffic in transit	Loss ofinformationCompromise ofmachineVulnerability to allotherthreats	Cryptographic checksums
Confidentiality	EavesdroppingontheNet TheftofinfofromserverTheftofdatafromclientInfoaboutnetworkconfiguration Infoaboutwhichclienttalks toserver	Loss of informationLossofprivacy	Encryption,webproxies
Denial ofService	KillingofuserthreadsFloodingmachinewithBogusrequests Filling up disk ormemory Isolating machine by DNSattacks	Disruptive AnnoyingPrevent user fromgettingworkdone	Difficulttoprevent
Authentication	Impersonation of legitimateusersDataforgery	Misrepresentation ofuser Belief that falseinformationis valid	Cryptographic techniques

Two typesofattacks are:

Passiveattacksincludeeavesdroppingonnetworktrafficbetweenbrowserandserverand gainingaccessto informationonaWeb sitethatissupposedtoberestricted.

Activeattacksincludeimpersonatinganotheruser,alteringmessagesintransitbetweenclientandserver,andalteringinformationonaWeb site.

5.3.2 WEBTRAFFICSECURITYAPPROACHES

OnewaytoprovideWebsecurityistouseIPSecurity.TheadvantageofusingIPSecisthat itis transparenttoendusers and applicationsandprovidesa general-purposesolution.

HTTP	FTP	SMTP
SSL/TLS		
TCP		

HTTP	FTP	SMTP
TCP		
IP/IPSec		

(a) Networklevel

IP

(b)TransportLevel

	S/MIME	PGP	SET
Kerberos	SMTP		HTTP
UDP	TCP		
IP			

(c) Application Level

Fig5.14: RelativeFacilitiesinthe TCP/IP **Location of SecurityProtocal Stack**

5.3.3 SECURESOCKETLAYER ANDTRANSPORTLAYERSECURITY

5.3.3.1 SSLArchitecture

SSLisdesignedto makeuseofTCPtoprovidea reliableend-to-endsecureservice.

TheSSLRecordProtocolprovidesbasicsecurityservicestovarioushigher-layerprotocols. In particular, the Hypertext Transfer Protocol (HTTP), which provides the transferserviceforWebclient/serverinteraction,canoperateontopofSSL.Threehigher-layerprotocolsaredefinedaspartofSSL:theHandshakeProtocol,TheChangeCipherSpecProtocol,andtheAlertProtocol.

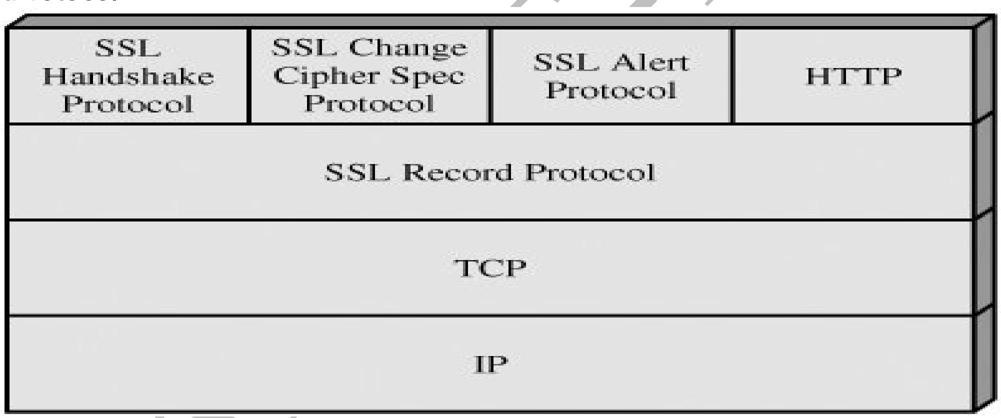


Fig 5.15:SSLProtocolStack

Two important SSL concepts are the SSL session and the SSL connection, which aredefinedinthespecificationas follows:

Connection:

A connection is a transport (in the OSI layering model definition) that provides a suitabletype of service. For SSL, such connections are peer-to-peer relationships. The connections aretransient.Everyconnectionisassociatedwithonesession.

Session:

AnSSLsessionisanassociationbetweenaclientandaserver.Sessionsarecreatedby the Handshake Protocol. Sessions define a set of cryptographic security parameters, whichcanbesharedamongmultipleconnections.Sessionsareusedtoavoidtheexpensivenegotiation ofnew securityparameters foreachconnection.

A sessionstateisdefined bythefollowingparameters

- Sessionidentifier
- Peer certificate
- Compressionmethod
- Cipherspec
- Mastersecret
- Isresumable

Aconnectionstate isdefined bythefollowingparameters:

- Serverand clientrandom
- Serverwrite MAC secret
- Client write MACsecret
- Serverwritekey
- Clientwritekey.
- Initializationvectors
- Sequence numbers

SSLRecord Protocol

TheSSL RecordProtocol providestwo servicesforSSL connections:

Confidentiality: The Handshake Protocol defines a shared secret key that is used forconventionalencryptionofSSLpayloads.

Message Integrity: The Handshake Protocol also defines a shared secret key that isusedtoforma messageauthentication code(MAC).

The diagram indicates the overall operation of the SSL Record Protocol. The RecordProtocol takes an application message to be transmitted, fragments the data into manageableblocks, optionally compresses the data, applies a MAC, encrypts, adds a header, and transmitsthe resulting unit in a TCP segment. Received data are decrypted, verified, decompressed, andreassembledandthendeliveredto higher-levelusers.

The first stepis fragmentation. Each upper-layermessage is fragmented into blocks of2^{14} bytes (16384 bytes) or less. Next, compression is optionally applied. Compression must belossless and may not increase the content length by more than 1024 bytes. In SSLv3 (as well asthe current version of TLS), no compression algorithm is specified, so the default compressionalgorithmisnull.

The next step in processing is to compute a **message authentication code** over thecompresseddata.

The final step of SSL Record Protocol processing is to prepend a header, consisting ofthefollowingfields:

- **ContentType(8bits):**Thehigherlayerprotocolusedtoprocesstheenclosedfragment.
- **MajorVersion(8bits):**IndicatesmajorversionofSSLinuse.ForSSLv3,thevalueis3.
- **MinorVersion(8bits):**Indicates minorversioninuse. ForSSLv3,thevalueis0.
- **CompressedLength(16bits):**Thelengthinbytesoftheplaintextfragment(orcompressedfragmentifcompressionisused).Themaximumvalueis2^{14}+2048.

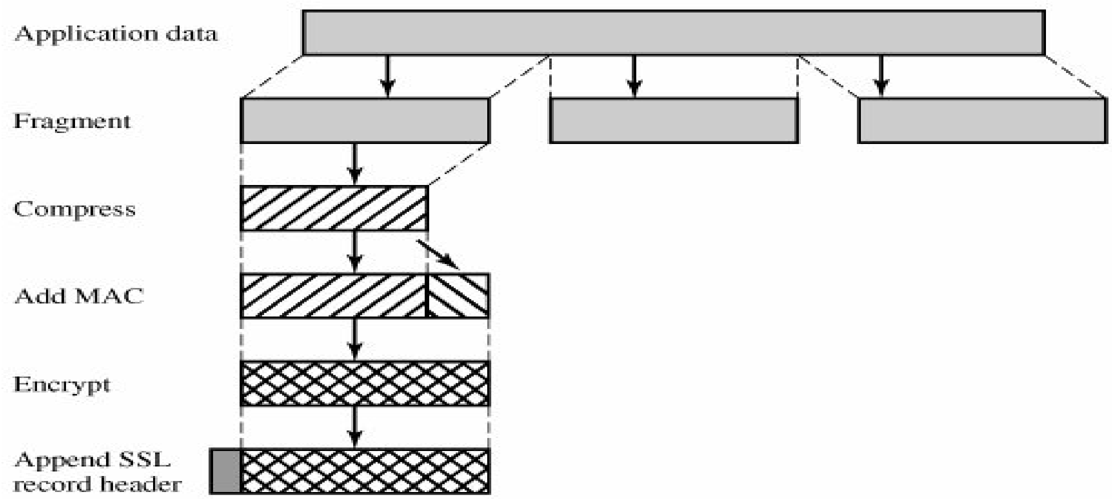


Fig 5.16: SSL Record Protocol

OperationChangeCipherSpec Protocol

Thisprotocolconsistsofasinglemessagewhichconsistsofasinglebytewiththevalue 1.

Alert Protocol

TheAlertProtocolisusedtoconveySSL-relatedalerts to thepeer entity.

Each message in this protocol consists oftwo bytes The firstbyte takes the valuewarning(1) or fatal(2) to convey the severity of the message. The second byte contains a codethatindicatesthe specificalert.

- **unexpected_message:**Aninappropriatemessage wasreceived.
- **bad_record_mac:**AnincorrectMAC was received.
- **decompression_failure:**Thedecompressionfunctionreceivedimproperinput(e.g.,unablet odecompressor decompresstogreater thanmaximum allowablelength).
- **handshake_failure:**Senderwasunabletonegotiateanacceptablesetofsecurityparametersg iventheoptionsavailable.
- **illegal_parameter:**Afieldinahandshakemessagewasoutofrangeorinconsistentwithotherfie lds.

Theremainderofthealerts isthefollowing:

- **Close notify:** Notifies the recipient that the sender will not send any more messageson this connection. Each party is required to send a close_notify alert before closingtherightside ofaconnection.
- **No certificate:** May be sent in response to a certificate request if no appropriatecertificateis available.
- **bad_certificate:** A received certificate was corrupt (e.g., contained a signature thatdidnotverify).
- **unsupported_certificate:**Thetype ofthereceived certificateis notsupported.
- **certificate_revoked:**Acertificate has beenrevokedbyitssigner.
- **certificate_expired:**Acertificate hasexpired.
- **certificate_unknown:**Someotherunspecifiedissuearoseinprocessingthecertificate,re nderingitunacceptable.

HandshakeProtocol

This protocol allows the server and client to authenticate each other and to negotiate anencryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSLrecord.The HandshakeProtocolisusedbeforeanyapplicationdatais transmitted.

TheHandshakeProtocolconsistsofaseriesofmessagesexchangedbyclientand server.

EstablishSecurityCapabilities

Thisphaseisusedtoinitiatealogicalconnectionandtoestablishthesecuritycapabilities that will be associated with it. The exchange is initiated by the client, which sends aclient_hellomessagewiththefollowingparameters:

1. Version: Thehighest SSLversionunderstoodbytheclient.

2. Random: A client-generated random structure, consisting of a 32-bit timestamp and28 bytes generated by a secure random number generator. These values serve as nonces andareused duringkey exchange to preventreplay attacks.

3. Session ID: A variable-length session identifier. A nonzero value indicates that theclient wishes to update the parameters of an existing connection or create a new connection onthissession.

4. CipherSuite: This is a list that contains the combinations of cryptographic algorithmssupportedby the client,indecreasingorderofpreference.

5. **Compression Method:** This is a list of the compression methods the client supports.After sending the client_hello message, the client waits for the server_hello message,whichcontainsthesameparametersastheclient_hellomessage.

ServerAuthenticationandKeyExchange

Theserverbeginsthisphasebysendingitscertificate;Thecertificatemessageisrequiredfor any agreed-onkey exchange methodexceptanonymousDiffie-Hellman.

Next, a server_key_exchange message may be sent if it is required. The certificaterequest messageincludestwoparameters: certificate_typeandcertificate_authorities.

ClientAuthenticationandKeyExchange

Ifthe serverhasrequested acertificate, the clientbegins thisphase by sendingacertificatemessage.Nextistheclient_key_exchangemessage,whichmustbesentinthisphase.

Finally,inthisphase,theclientmaysendacertificate_verifymessagetoprovideexplicitverificationofa clientcertificate.

SSLHandshakeProtocolMessageTypes

MessageType	Parameters
hello_request	null
client_hello	version,random,sessionid,ciphersuite,compression method
server_hello	version,random,sessionid,ciphersuite,compression method
certificate	chainofX.509v3certificates
server_key_exchange	parameters,signature

SSLHandshakeProtocolMessageTypes	
MessageType	**Parameters**
certificate_request	type,authorities
server_done	null
certificate_verify	signature
client_key_exchange	parameters,signature
finished	hash value

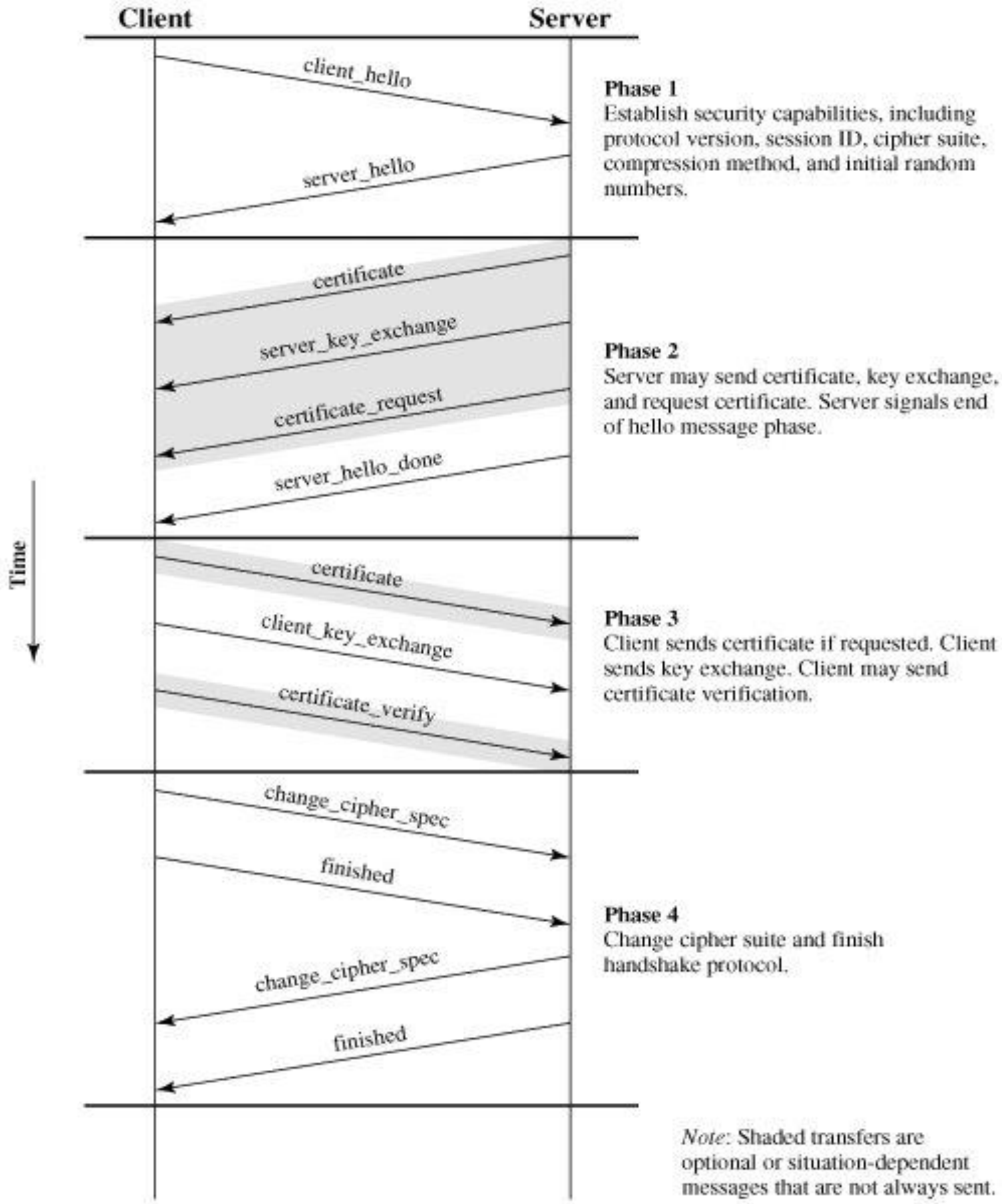


Fig5.17:HandshakeProtocolAction

Finish

Thisphasecompletesthesettingupofasecureconnection.Theclientsendsa change_cipher_specmessageand copiesthependingCipherSpecintothecurrent CipherSpec.Theclientthenimmediately sends thefinished messageunderthenewalgorithms, keys, and secrets.

Cryptographic ComputationsMasterSecretCreation

Thesharedmastersecretisaone-time48-bytevalue(384bits)generatedforthissessionby means ofsecurekeyexchange.The creationisintwostages.

First,apre_master_secretisexchanged.Second,themaster_secretiscalculatedbybothparties .Forpre_master_secretexchange,thereare twopossibilities:

- **RSA:**A48-bytepre_master_secretisgeneratedbytheclient,encryptedwiththeserver's public RSA key, and sent to the server. The server decrypts the ciphertext usingitsprivatekeytorecoverthepre_master_secret.

- **Diffie-Hellman:** Both client and server generate a Diffie-Hellman public key. After theseare exchanged, each side performs the Diffie-Hellman calculation to create the sharedpre_master_secret.

5.3.4 SECUREELECTRONICTRANSACTION

SETisanopenencryptionand securityspecificationdesignedto protect credit cardtransactionson the Internet.

SETisnot itselfapaymentsystem. Ratherit isasetofsecurity protocolsandformatsthat enablesusers to employtheexistingcredit cardpaymentinfrastructure onanopennetwork,such as the Internet,inasecure fashion.

SET providesthreeservices:

- Providesasecurecommunicationschannelamongallpartiesinvolvedinatransaction
- Provides trustbytheuseofX.509v3 digitalcertificates
- Ensuresprivacybecausetheinformationisonlyavailabletopartiesinatransactionwhen andwherenecessary

5.3.4.1 KeyFeaturesofSET

- Confidentialityofinformation
- Integrity ofdata
- Cardholderaccount authentication
- Merchant authentication

5.3.4.2 SETParticipants

- **Cardholder:**Acardholderisanauthorizedholderofapaymentcard(e.g.,MasterCard,Visa)tha thasbeen issuedby anissuer.
- **Merchant:**Amerchantisapersonororganizationthathasgoodsorservicestoselltothecardhol der.
- **Issuer**:Thisisafinancialinstitution,suchasabank,thatprovidesthecardholderwiththepaymen tcard.
- **Acquirer**:Thisisafinancialinstitutionthatestablishesanaccountwithamerchantandprocesse spaymentcardauthorizationsandpayments.

- **Paymentgateway**: This is a function operated by the acquirer or a designated thirdparty thatprocessesmerchantpaymentmessages.
- **Certification authority (CA):** This is an entity that is trusted to issue X.509v3 public-keycertificates forcardholders,merchants,andpaymentgateways.

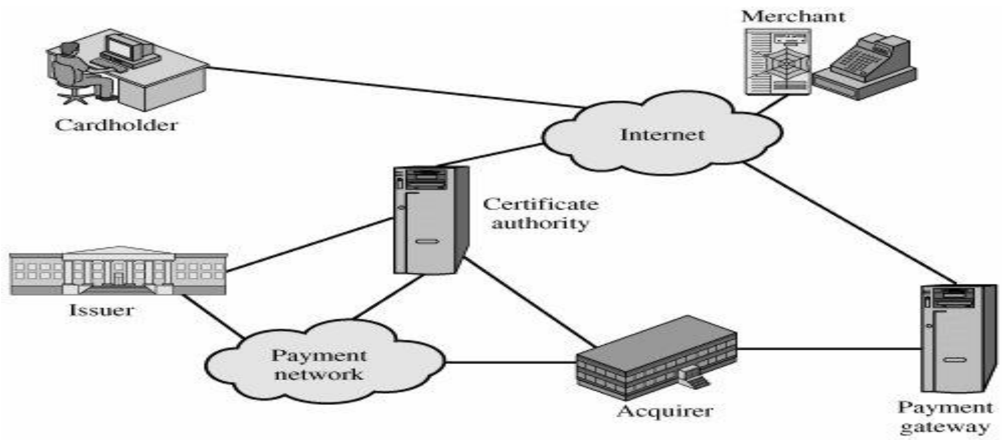


Fig5.18:SecureElectronicCommerceComponents

5.3.4.3 SETTransaction

- **Customeropensaccount:**Thecustomerobtainsacreditcardaccount,suchasMasterCardor Visa,withabankthatsupportselectronic paymentandSET.
- **Customer receivesa certificate:**After verification the customer receives X.509V3,digital certificate which is signed by the bank. This certificate verifies the customer‟s RSApublickey andexpiration date.
- **Merchantshavetheirowncertificates:**
  - ✓ Merchantswhoacceptscardneedtohave2certificatesfor2publickeysownedby them.
  - ✓ Onecertificateisusedforsigningofmessageandtheotherisusedforkeyexchange.
  - ✓ Themerchantsalsoneedthecopyofpaymentgateway‟spublickeycertificate.
- **Customerplacesan order:**
  - ✓ Thecustomerplacestheordercontainingthelistofitemstobepurchasedtothemerchant .
  - ✓ The merchantreturnstheorderformhavingtheitems,price,totalpriceandordernumber.
- **Merchantisverified:**Themerchantalongwiththeorderformsendsitscertificatecopy.The customercan verifythe same.
- **Orderand paymentaresent:**
  - ✓ Thecustomersendsorderandpaymentinformationintothemerchantalongwithcustomer‟scertificate.
  - ✓ Thisisorderconformationoftheorder form.
  - ✓ Thepaymentcontainsthecarddetails.Thisisencrypted,soitcannotbereadbythe merchant.
  - ✓ Thecertificate sentcanbeverified bythemerchant.

- **Merchantrequestspaymentauthorization:**Themerchantsendsthepaymentinformation to the payment gateway. The merchant requests for authentication of thecustomer,creditlimit,validity.
- **Merchantconfirmsorder:**Themerchantsendsconformationoftheordertothecustomer.
- **Merchantprovidesgoodsor service**
- **Merchantrequests payment**

5.3.4.4 DualSignature

The purpose of the dual signature is to link two messages that are intended for twodifferent recipients. In this case, the customer wants to send the order information (OI) to themerchant and the paymentinformation (PI) to the bank. The merchant doesnot need to knowthe customer's credit card number, and the bank does not need to know the details of thecustomer'sorder.

The customer takes the hash (using SHA-1) of the PI and the hash of the OI. These twohashesarethenconcatenatedandthehashoftheresultistaken.Finally,thecustomerencrypts the final hash with his or her private signature key, creating the dual signature. Theoperationcan be summarizedas

$DS=E(PR_c,[H(H(PI)|| H(OI)])$

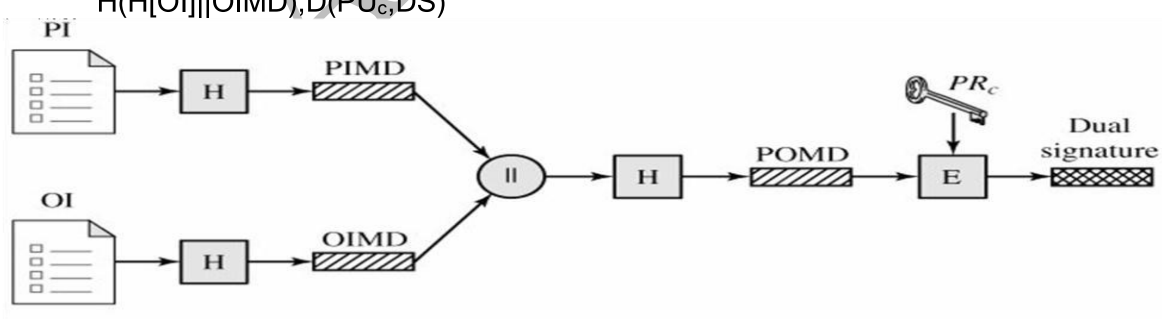
Where PR_c is the customer's private signature key. Now suppose that the merchant is inpossession of the dual signature (DS), the OI, and the message digest for the PI (PIMD). Themerchantalsohasthepublickeyofthecustomer,takenfromthecustomer'scertificate.Thenthemerchantcancomputethequantities

$H (PIMS||H[OI]);D(PU_c,DS)$

Where PU_c is the customer's public signature key. If these two quantities are equal, thenthemerchanthasverifiedthe signature.

Similarly, if the bank is in possession of DS, PI, the message digest for OI (OIMD), andthecustomer'spublickey,thenthebankcan compute

$H(H[OI]||OIMD);D(PU_c,DS)$



PI= Payment information PIMD=PImessagedigestOI
= Orderinformation OIMD =OImessagedigest
H=Hashfunction(SHA-1) POMD
=PaymentordermessagedigestII=Concatenation E=Encryption(RSA)

PR_C=Customer‟sprivatesignaturekey

Payment Processing

- Purchase request
- Paymentauthorization
- Paymentcapture

PurchaseRequest

Before the Purchase Request exchange begins, the cardholder has completed browsing,selecting, and ordering. The end of this preliminary phase occurs when the merchant sends acompleted orderformtothecustomer.

The purchase request exchange consists of four messages: Initiate Request, InitiateResponse,Purchase Request,andPurchase Response.

- verifiescardholdercertificatesusing CAsigs
- verifies dual signature using customer's public signature key to ensure order hasnot been tampered with in transit & that it was signed using cardholder's privatesignature key
- processes order and forwards the payment information to the payment gatewayfor authorization(describedlater)
- sends apurchaseresponsetocardholder

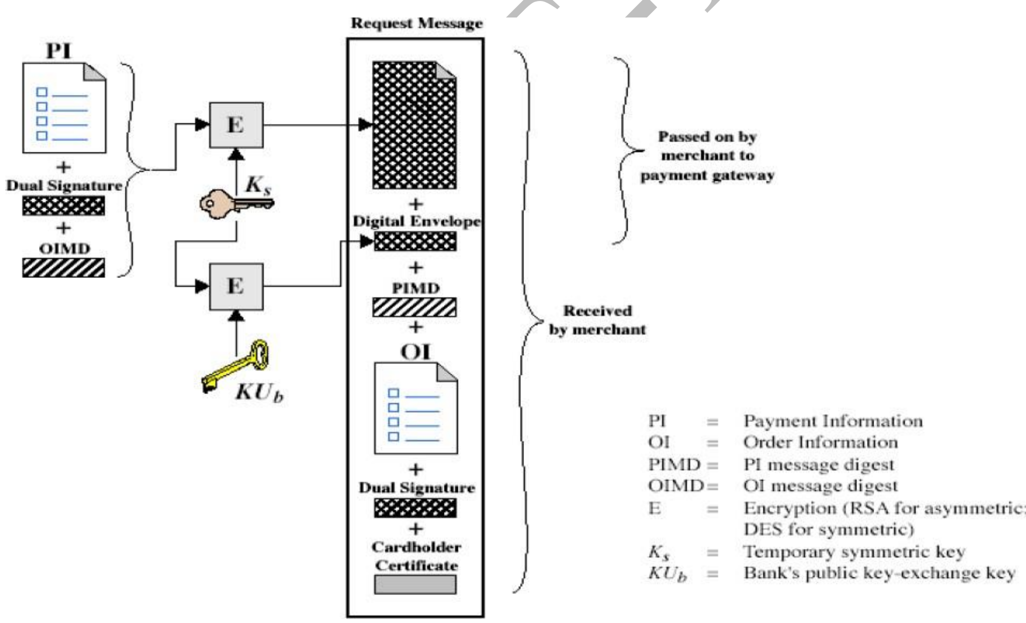


Fig 5.19:PurchaseRequest–Customer

PaymentAuthorization

The paymentauthorization ensures that thetransaction was approved by the issuer.Thisauthorizationguaranteesthatthemerchantwillreceivepayment;themerchantcantherefore provide the services or goods to the customer. The payment authorization exchangeconsistsoftwomessages:AuthorizationRequestandAuthorizationresponse.

- Verifiesallcertificates

- Decryptsdigitalenvelopeofauthorizationblocktoobtainsymmetrickey&thendecryptsauthorizationblock
- Verifiesmerchant'ssignatureon authorizationblock
- Decryptsdigitalenvelopeofpaymentblocktoobtainsymmetrickey&thendecryptspayment block
- Verifiesdualsignature onpaymentblock
- VerifiesthattransactionIDreceivedfrommerchantmatchesthatinPIreceived(indirectly)fromcustomer
- Requests&receives an authorization from issuer
- Sendsauthorization response backtomerchant

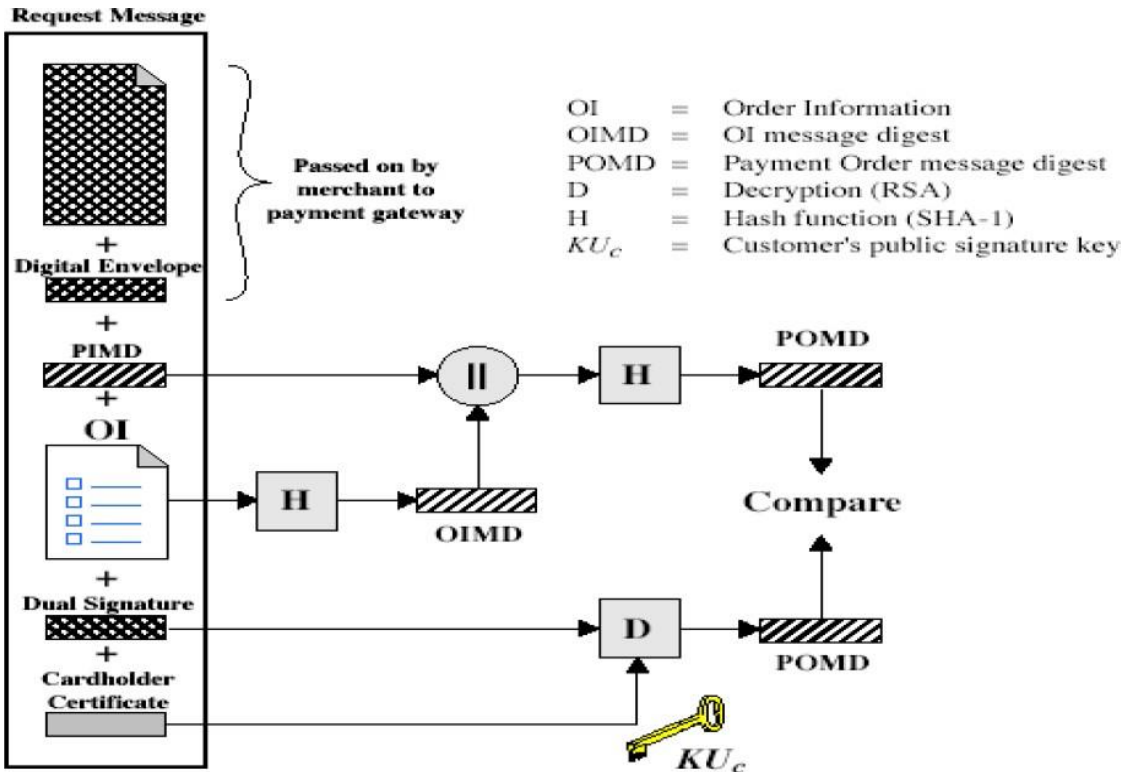


Fig 5.20: PurchaseRequest–Merchant

PaymentCapture

Toobtainpayment,themerchantengagesthepaymentgatewayinapaymentcapturetransaction ,consistingofacapture requestandacaptureresponsemessage.

- Merchantsends paymentgatewayapaymentcapture request
- Gateway checksrequest
- Then causesfunds to be transferred to merchantsaccount
- Notifiesmerchantusingcaptureresponse

Firewalls

Firewall are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet,

All messages entering or leaving the intranet pass through the firewall, which examines each message & blocks those do not meet the specified security criteria.

Firewall is a term used for a "barrier" between a network of machines & users that operate under a common security policy & generally trust each other & the outside world.

Related terminology

1. Rules → AXS GUARD
2. Policies
3. Security level
4. Firewall Rights
5. Firewall GUI : → a) Create, Edit
   b) Assign Firewall policies
   c) Consult Firewall logs & Firewall status.

Characteristics

1. Traffic from inside to outside & vice versa must pass through the firewall.
2. Only authorized traffic will be allowed to pass
3. Firewall itself is immune to penetration

A Techniques to control access

a) Service control - Type of Internet service that can be accessed.

b) Direction control - Particular service gets initialized & allowed to flow

c) User control - All the users who have access to the other side of Firewall

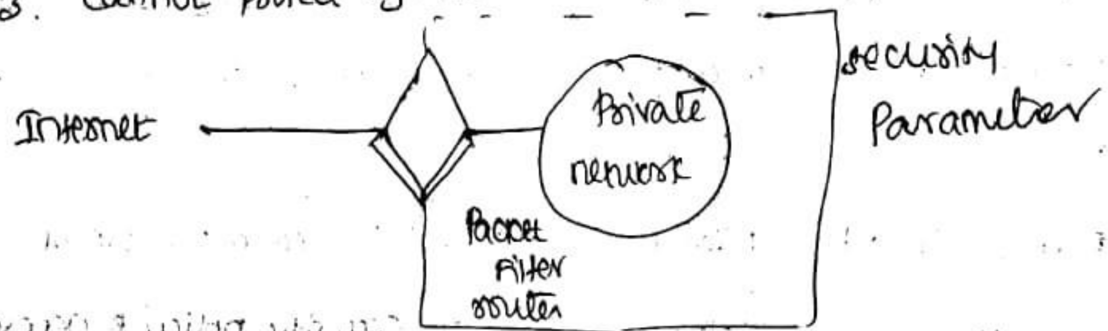
d) Behaviour control - Particular services are used for an Application

Limitations

1. Firewall can be bypassed

2. Does not protect against internal threats

3. cannot protect against transfer of virus infected programs



Types

1. Packet Filter.

2. Application-level gateway.

3. circuit-level.

Packet [static filtering. (works on network layer].

A packet filter router applies a set of rules of each incoming & outgoing IP packet & then forwards or discards the packet.

Based on

a) Source IP address

b) Destination IP "

c) Source & destination

d) IP protocol field.

e) Interface.

[A Firewall technique used to control n/w access by monitoring outgoing & incoming packets & allowing them to pass or halt based on the transport level address]

source & destination IP addresses, protocols & ports.

Advantages

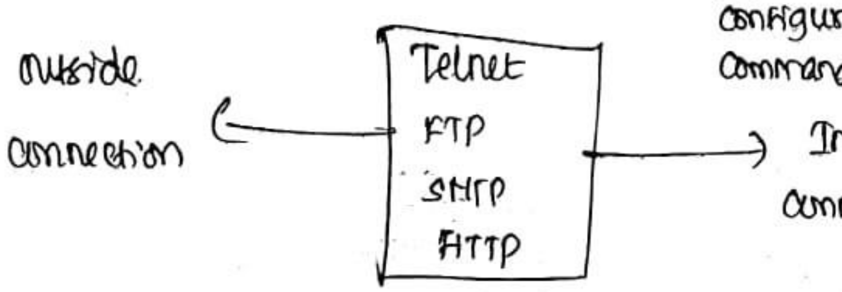
1. simplicity 2. Fast operations 3. Transparent to users

Limitations

1. Firewalls do not examine upper layer data. since limited information is available to firewall its functional is limited.

2. It does not support advanced user authentication schemes.

Application-level



Eg: SMTP application proxies can be configured to allow only certain commands like mailfrom: rcpt to: etc to pass through firewall & block other commands.

* It is also called a proxy server & it acts as a relay of Application level traffic.

* The user contacts the gateway using a TCP/IP application such as Telnet or FTP & gateway asks the user for the name of the remote host to be accessed.

* when userid is provided, the gateway contacts the application on the remote host & relays that TCP segment between two end points.

* gateway not implement proxy code, the service is not supported.

Ad
1. More secure than packet
2. Easy to log & audit all

Disadvantage:
1. Additional processing overhead on each Application

3) Circuit-level [works on session layer]

Stand alone system.

Not End - End TCP connection

Two connection
1) one between itself & an inner host
2) one between itself & outer host

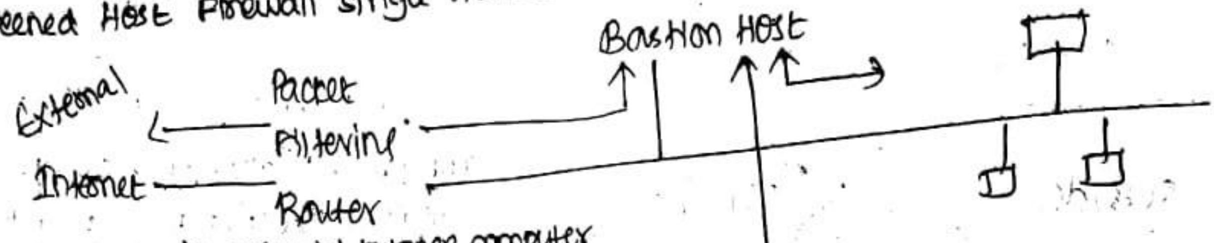
Firewall Designs

Bastion Host

System identified by the firewall administration as a critical strong point in n/w security.

Configurations

1. Screened Host Firewall single homed Bastion.



[A bastion host is a special purpose computer on a network specifically designed & configured to withstand attacks. The computer generally hosts a single app for eg. a proxy server, and all other services are removed or limited to reduce the threat to computer.]

Information Server

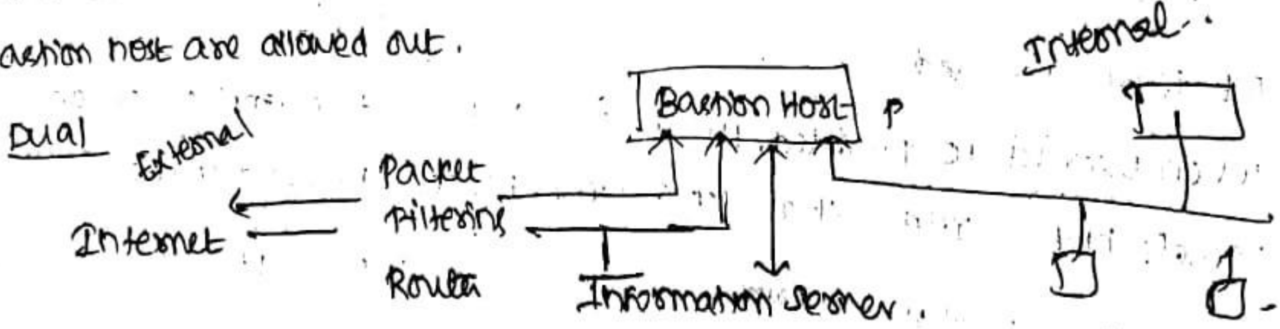
Router is configured so that

(External n/w)

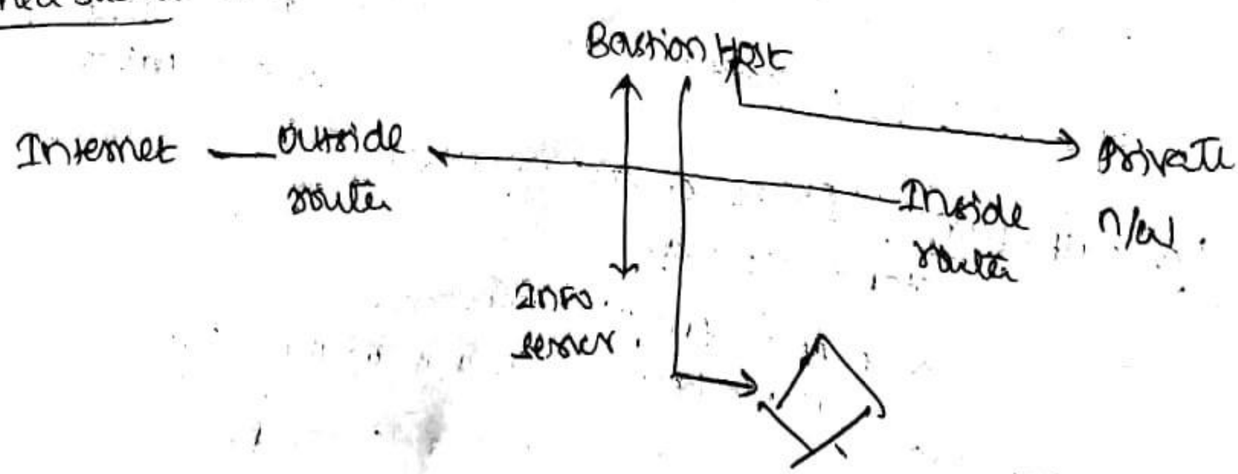
a) For traffic from the Internet, only IP packets destined for the Bastion host are allowed in Internal n/w

b) For traffic from the internal network, only IP packets from the bastion host are allowed out.

Dual



Screened subnet



1) Two filtering router

1. one between bastion host & Internet

2. one between bastion host & Internal n/w

2) Both Internet & the internal network have access to the host on the screened subnet but traffic across the screened subnet is blocked.

Ad: outer → screened subnet to Internet.

Inside → screened subnet to Internal n/w

Packet Filters	Application-level gateway	circuit-level gateways.
1. simplest & least secure firewall mech.	1. Most secure approach	More secure than packet, but was not as secure as application
2. Many routers provide this function.	2. Unique pgm for each application	Relay TCP connection
3. Passes or rejects packets based on rules	3. Good for authentication, logging.	Permission granted by port address.
4. Hard to manage.	4. Not always transparent to users.	No application-level checking
	Used for e-mail, FTP, Telnet, www	

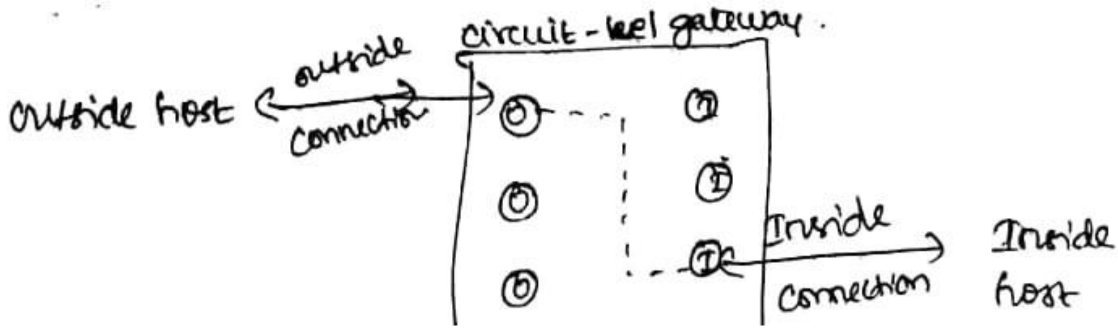
Application gateways → Application, Presentation, Session

circuit gateways → Transport

Packet filtering → Network

MAC layer firewall → Data link

Physical.



Routers

Routers route packets of data from one network to another. Some routers even control the internet's infrastructure.

Gateways

Gateways are the portals that computers use to connect to the internet. One computer can be a gateway for others which is made possible through ICS (Internet Connection Sharing)

Differences

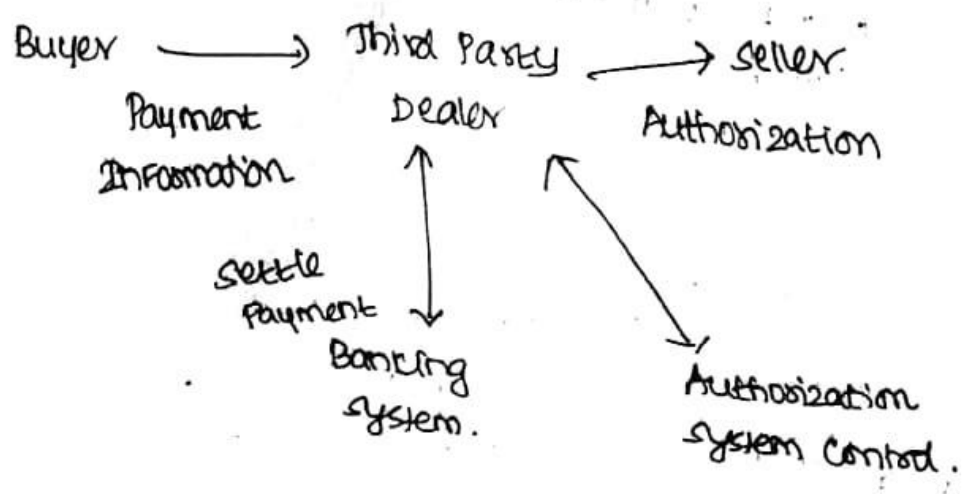
Router coordinates data transfer from one computer to another within a n/w & to other n/w.

Gateway is any device specifically designed to provide all the computer in a n/w with access to WWW.

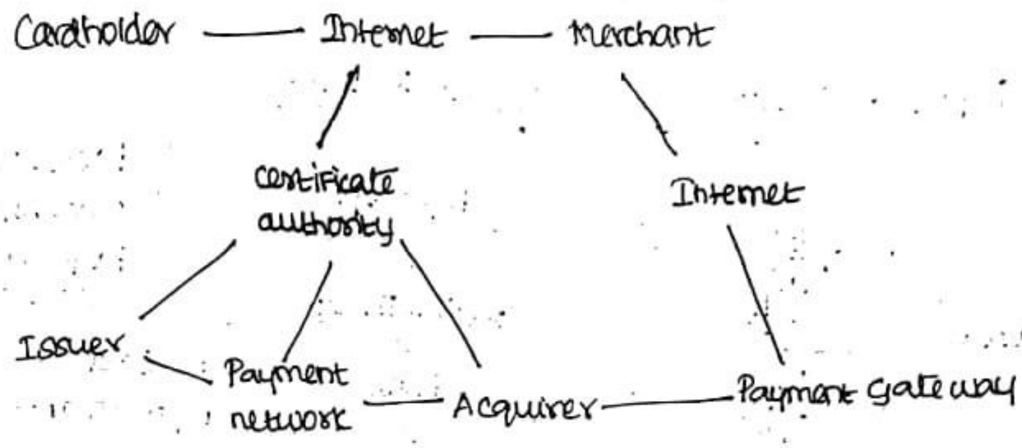
SET For E-Commerce Transactions

The Secure Electronic Transaction (SET) is the credit card Payment Protocol that securely transfers the money from customer to the merchant with high integrity.

Structure



SET Components



Requirements

* Payment Confidentiality
* Ordering Confidentiality
* Transmitted Data Integrity
* Transmitted Data privacy
* Card holder Buyer Authentication
* Seller Authentication

Dual signature

Authorizing that particular Payment is intended for particular order.

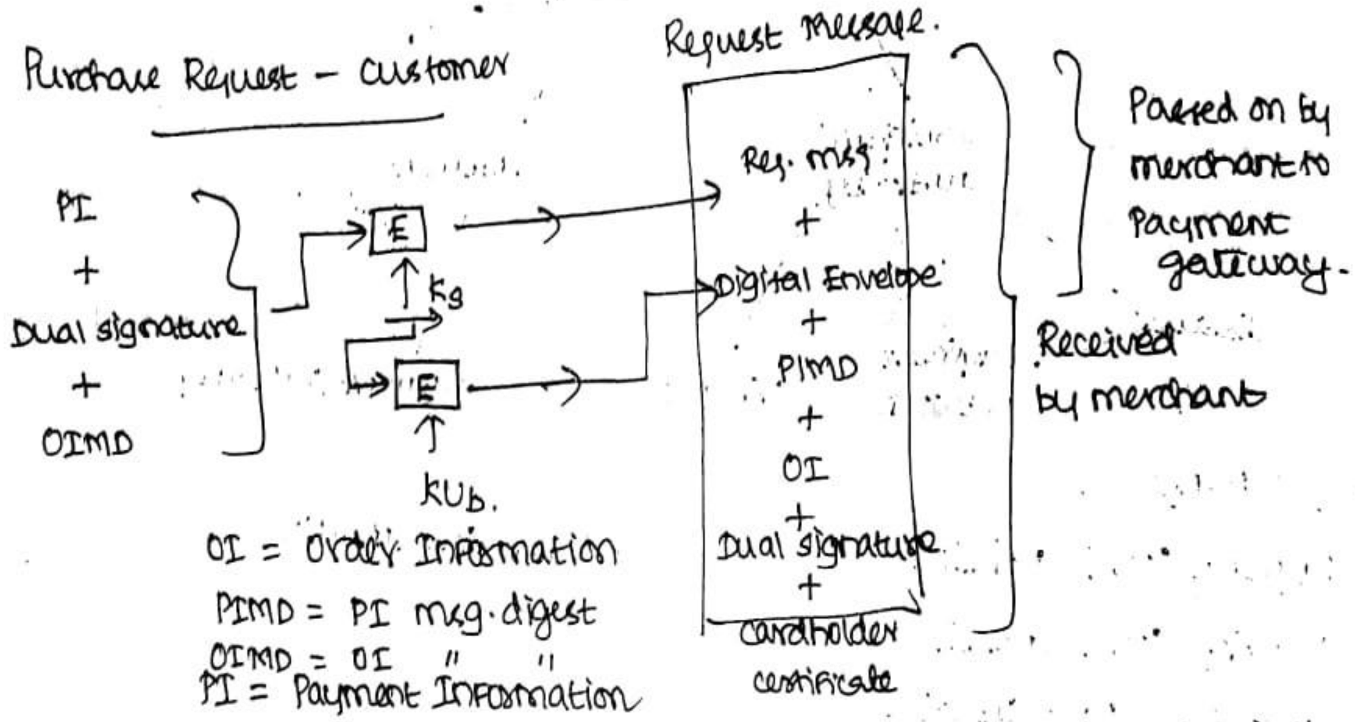
1. Payment Information (PI) for bank
2. Order Information (OI) for merchant

"The purpose of dual signature is to link two messages that are intended for two different recipients".

Customer wants to send the Order Information (OI) to the merchant & the payment Information (PI) to the bank. Merchant does not need to know the customer's credit-card number, and the bank does not need to know the details of the customer's order.

The Dual signature is the encrypted MD (with the customer's secret key) of the concatenated MD's of PI and OI. The dual signature is sent to both the merchant and the bank. The protocol arranges for the merchant to see the MD of the PI without seeing the PI itself, and the bank sees the MD of the OI but not the OI itself. The dual signature can be verified using the MD of the OI or PI. The MD does not reveal the content of the OI or PI, thus privacy is preserved.

Purchase Request – customer



OI = Order Information

PIMD = PI msg. digest

OIMD = OI " "

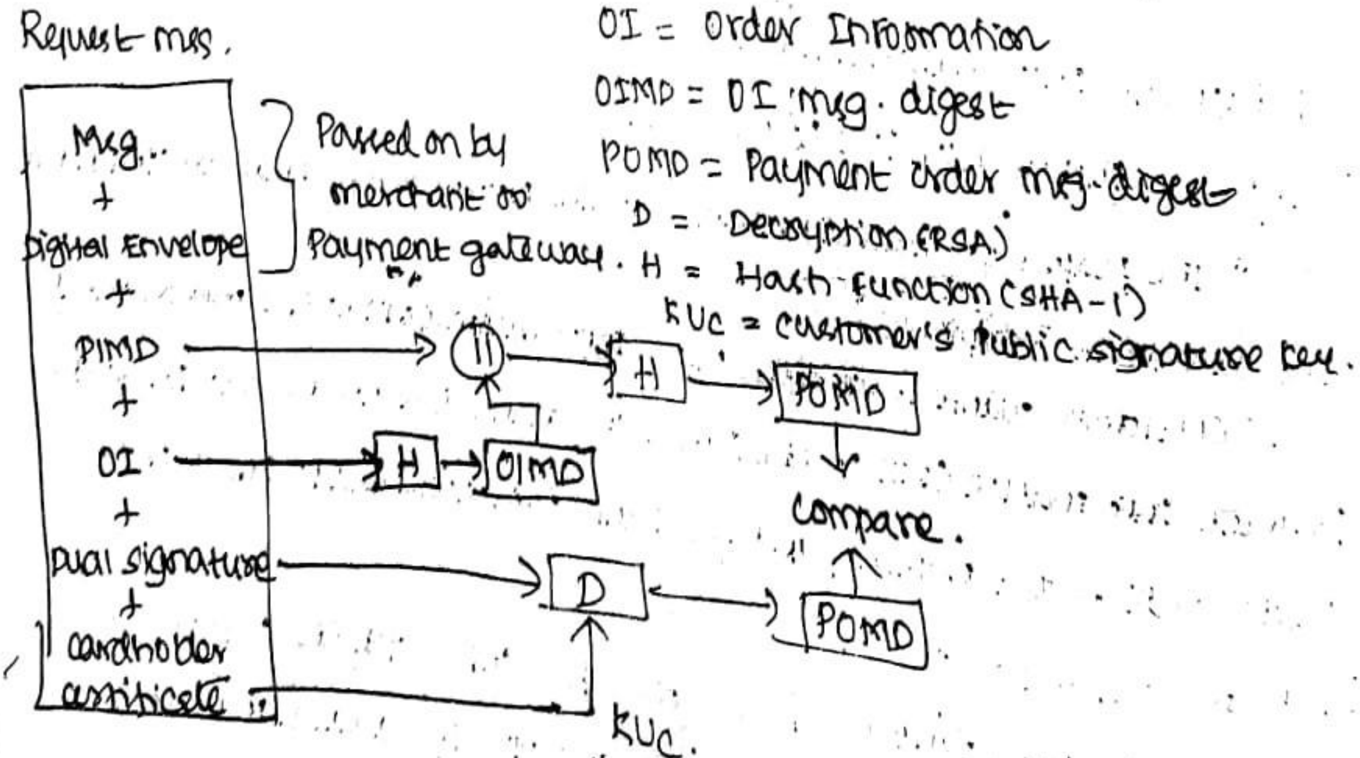
PI = Payment Information

E = Encryption (RSA for asymmetric ; DES for symmetric).

Ks = Temporary symmetric key.

KUb = Bank's public key-exchange key.

Purchase Request – Merchant



OI = Order Information

OIMD = OI msg. digest

POMD = Payment order msg. digest

D = Decryption (RSA)

H = Hash function (SHA-1)

KUc = customer's public signature key.

1. verifies cardholder certificates using CA

2. verifies dual signature using customer's Public Signature Key to ensure order has not been tampered with in transit & that it was signed using cardholder's Private Signature key.

3. Processes order & forward the Payment information to the Payment gateway for authorization

4. Sends a Purchase response to Cardholder.

Intrusion Detection

Intruders:

1. Masquerader: Person or a system that is not authorized to use the computer but penetrates a system and access controls to exploit an authorized users account.

2. Misfeasor: An authorized user who misuses the privileges by accessing data, programs or resources for which such access is not authorized.

3. Clandestine User: An individual who seizes supervisory control of the system to evade auditing & access controls or to suppress audit collection.

Techniques

The main aim of any intruder is to gain access to a system or to increase the range of privileges accessible on a system.

The Intruder needs to know certain details that are protected in the system. This can be done by gaining access to the password file for each authorized user.

Techniques (intrusion)

→ Target acquisition & Information gathering

→ Initial access

→ Privilege escalation

→ Covering tracks

Password guessing

→ one of the most common attacks

→ attackers knows a login (from email / web page etc.)

→ try default pswds shipped with systems

→ try all short passwords

→ then try by searching dictionaries of common words

→ Intelligent searches try passwords associated with the user (variations on names, birthday, phone, common words / interfaces)

Password capture

→ watching over shoulder as password is entered

→ using a trojan horse program to collect

→ extracting recorded info after successful login (web history / cache, last number dialed etc)

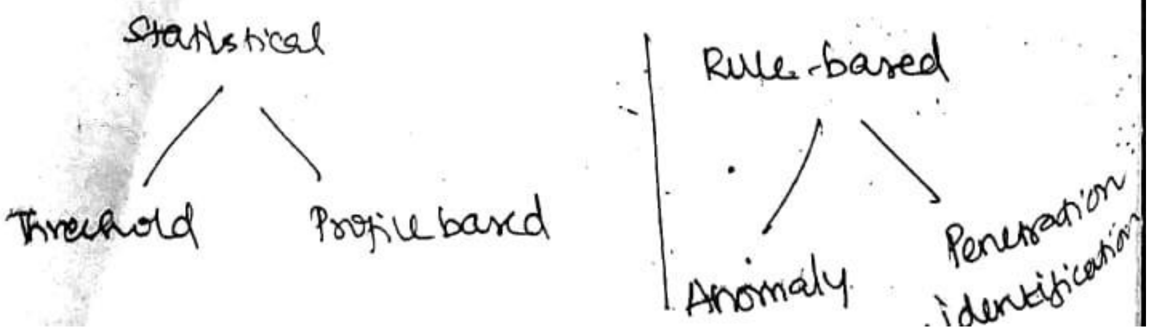
Approaches to [illegible] Detection

1. Statistical Anomaly Detection:

Involves collection of data relating to the behaviour of authorized users over a period of time.

2. Rule based detection

Involves an attempt to define a set of rules that can be used to decide that a given behavior is that of an intruder.



Statistical

a) Threshold detection

Involves counting the number of occurence of a specific event type over an interval of time.

b) Profile based

Profile of the activity of each user is developed and used to detect the changes in the past behaviour of individual accounts.

Metrics used

1. Counter — keeps a count of certain event types is kept over a particular period of time.
2. Gauge — Used to measure the current value of some entity
3. Interval timer — Notes the length of time between two related events
4. Resource utilization — Quantity of resource that is consumed during a period of time.

Rule Based

a) Rule based Anomaly Detection

→ Historical Audit records are analyzed to identify usage pattern & to generate automatically rules that described patterns.

→ Based on observing past behaviour & assumes that the future will be like the past.

b) Penetration Identific

Rules for identifying known penetration that would exploit known weaknesses.

Rules specific to machine & O.S

Audit Records → Fundamental tool

1. Native Audit record : Any multiuser os includes accounting s/w that collects the information on user activity
2. Detection specific audit : Facility that can be implemented by generated audit records.

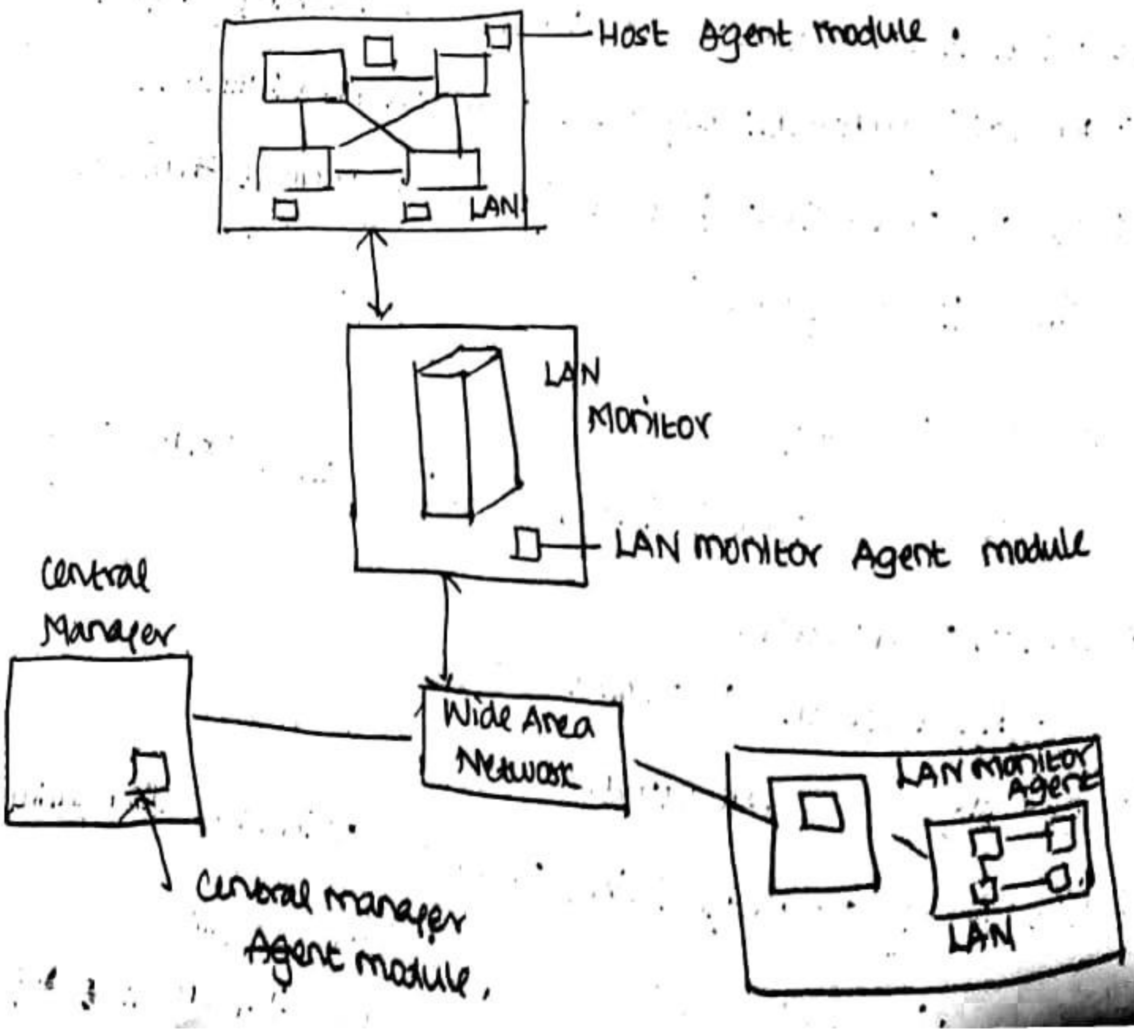
Fields

1. Subject → Initiate an action Eg: terminal user, group of users
2. Action → Performed by subject Eg: login, read, execute
3. Object → Entities on which Action is performed Eg: files, Programs, rec
4. Exception condition → Raised
5. Usage → Quantitative elements amount list of resources used
6. Time stamp → Action took place.

Subject	Action	Object	Exception	Resource	TS
Shyam	Read	<shyam> Hello.Exe	0	CPU=0005	1106872165

Distributed Intrusion Detection method

Architecture of the Intrusion Detection system for Distributed Network is defined by



Three components

1. Host Agent module 2. LAN monitor Agent module 3. Central Manager Agent module.

1. Each host system is configured with the agent module to monitor and collect the audit information.

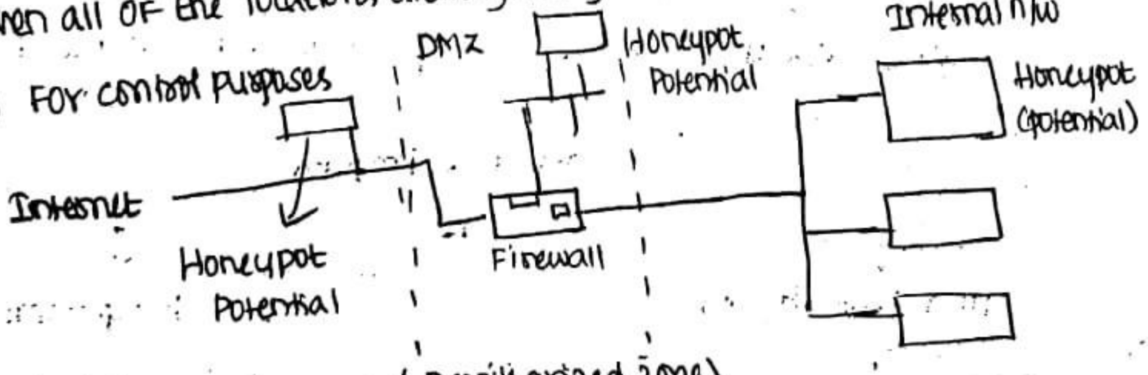
2. Each LAN network has LAN monitor Agent module that analyzes the user agent module & reports the same to central Manager. ~~Agent module~~

3. The central manager Agent module collects the report from LAN monitor agent module & compares the reports with its predefined reports to detect the intrusion.

Honeypots

Another Intrusion Detection system. Honey pot systems are decoy servers or systems setup to gather information regarding an attacker or intruder into your system.

Honeypots can be setup inside, outside or in the DMZ of a Firewall design or even all of the locations although they are most often deployed inside of a firewall for control purposes.



∴ DMZ - (Demilitarized zone)
↓
Add an additional layer of security to an LAN

It Designed, to divert an attacker from accessing critical systems. It collects information about the attackers activity, then it encourages the attackers to ~~say~~ stay on the system long enough for administrator to respond.

Viruses and Related threats

	Virus	Worms
How does it infect a computer system?	It inserts itself into a file or executable program	It exploits a weakness in an application or OS by replicating itself
How can it spread?	It has to rely on users transferring infected files/programs to ~~server~~ other computer systems	It can use a n/w to replicate itself to other computer system without user intervention
Does it infect files?	Yes, it deletes or modifies files. sometimes a virus also change the location of files.	Usually not. Worms usually ~~not~~ monopolize the CPU & memory.
Whose speed is more?	Virus is slower than worm	Worm is faster than virus. Eg: code red worm affected 3 lak PC in 14 hrs.
Definition	The virus is the program code that attaches itself to application program & when application prgm run it runs along with it.	The worm is code that replicate itself in order to consume resources to bring it down.

Malicious Program

- Need host Program
  - Trapdoor
  - Logic bombs
  - Trojan horses
  - Viruses
- Independent
  - Worms
  - Zombie.

1. Backdoor:

Backdoor is also called a trapdoor and it is a secret entry point in the program to gain access with program output. It completely damages the system with unimaginable number of intruders use backdoor to give access to the programs.

2. Logic bomb:

Piece of software code that is embedded in the program. events

i) Destroy some activities in the system
2) Embedded code sends the specified information to attacker.
3) Inform the attacker, that specific user has logged into machine for attack to begin.

3) Trojan horse

Useful Application program to user view contains hidden piece of code that causes damage to system activity.

a) Deletes the user file.
b) can Indirectly perform many damages that are not possible even by an unauthorized users.

4) Virus: Piece of software code or package that searches other programs or files & destroys them by inserting a copy of itself into the system.

5) Worms: Copies itself into the new machine & makes the replication.

Actively transmit itself in the infection process.

a) Self carried worms: actively transmit itself in the infection

b) Second channel worms: Secondary communication channel to complete the infection

c) Embedded worm: Makes itself as a copy in the normal communication channel either by appending or replacing the original message.

6) Zombie: Secretly captures another internet attached system and then uses that machine to embed the attacks that are very difficult to identify. DOS

7) Bacteria: Replicates by affecting the processor, memory & CPU disk.

Characteristics of Virus / Nature of Virus

Dormant ⟶ Propagation ⟶ Triggering ⟶ Execution

Dormant: Idle state & activated by some event date, capacity of disk

Propagation: Keeps identical copy of itself into another pgm. Every infected pgm. will have the virus which is now in the propagation phase.

Triggering: Makes the virus to be activated to perform its desired function for which this virus is designed.

Execution: Virus to be executed in the target pgm. Outcome of virus i execution may be data destruction

Types of virus / Classification

1. Parasitic virus [Attaches to Executable Files & replicates]

2. Memory Resident virus : Lodges in main memory as a part of resident system Pgm.

3. Boot sector virus : Infects a master boot record or spreads a system is booted from disk.

4. Stealth virus : [Hide from detection by antivirus s/w]

5. Polymorphic virus : mutates with every infection ["signature" different antivirus cannot detect]

6. Email virus : "Melissa virus" Spreading Email virus made use of a MS word attachment

7. Metamorphic virus : [similar to Polymorphic, gets more power after each infection
w. rewrites its own code.

8. Macro virus : Infect MS word document.

Worms : Remote Execution capability : [worms executes a copy of itself on another system.
Remote login capability : A worm logs onto a remote system as a user & then uses commands to copy itself from one system to another.

Virus Countermeasures

1. Antivirus methods — Virus Detection, Virus Identification, Virus Removal or Deletion
2. Techniques for Antivirus.

Generation of Antivirus S/W

1. First generation - Basic virus scanner [Uses virus signature & detect only specific virus
2. Second - Heuristic : Not using signature & uses heuristic rules to find.
3. Third - Activity : Identify not by structure by the action performed.
4. Fourth - Full Featured : Antivirus s/w techniques that are parallely executed [control capability to restrict reproduce in the system]

Techniques for Antivirus method [To develop antivirus]

1) Generic Decryption method
2) Digital Immune method
3) Behaviour Blocking s/w

1) General Decryption Method :

Detect & Fastly scan the Polymorphic virus From the system. Generally the virus must decrypt itself to be activated in the system.

1. CPU Emulator : (interprets the instruction in executable file rather than on Processor)

2. Virus signature scanner [Scans the code For known virus signature]

3. Emulation control module [controls execution OF virus infected code]

2) Digital Immune method : [Designed For Internet based virus]

Information From virus Infected client m/c

↓

Administrative m/c.

↓

Received by virus Analysis m/c Engine

↓

Analyze the virus Behaviour

↓

Analyze the virus structure.

↓

Extract the virus signature.

↓

Derive virus description to Individual user.

3) Behaviour Blocking method : [Integrates with OS OF a computer & monitors For malicious actions.

1) Modifying critical system settings such as startup

2) Operating, view, delete or modify Files.

3) Initiating N/W communications.

4. Formatting disk.

5) Modify Exec Files.

6. scripting OF e-mail & msg to client.

Practical Implementation OF cryptography

1. Code book : Code replaces word or phrase with character & symbol

∧	%	@	∞	⌫	≠
and	mrg.	Night	Tanks	Bombrun	Tmw.

P. T ⇒ Tanks and bomb run tmw. Night.

cod ⇒ ∞ ∧ ⌫ ≠ @ .

2. Nomenclators : Use elements OF substitution cipher.

a	b	c
0	≠	∧ . .

3. Smart cards

For building entry system, ATM's etc.

4. Biometrics : Authenticating an individual by personal characteristics.

Token is unique [Replace password - based authentication].