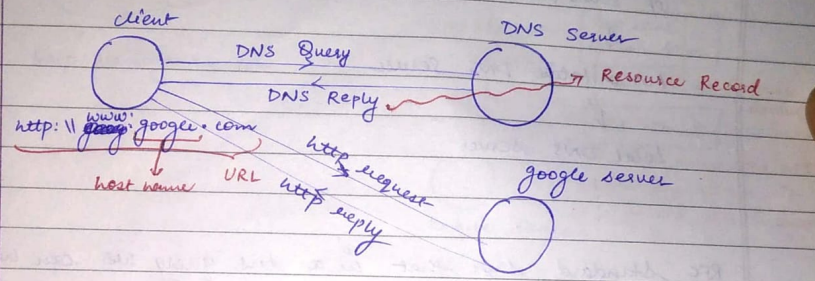
Buy alias hosts as well to decrease ambiguity & duplicacy.

03/06/18

Date : 03 / 06 / 18
Page No.

DNS : Domain Name Service.

DNS Server is used for mapping host names to IP addresses or vice versa.



→ DNS server is a database containing all IP addresses of different servers in the world.

DESIGN OF DNS SERVER:

1.) DNS Servers should be placed in heirarchy so that searching time will be less.

(There are so many entries in DNS server. If placed simply, then search time is very high)

2.) DNS servers should be placed in different geographical areas so that the propogation time is less.

(If DNS server is placed only in UK & people from India will require high P.T. to access whereas people from UK can access it easily
∴ many DNS servers should be placed across the globe)

Date : / /
Page No.

Heirarchy of DNS Server :

Root Server

↓

Top Level domain server

↓

Authoritative DNS Server

↓

Local DNS Server

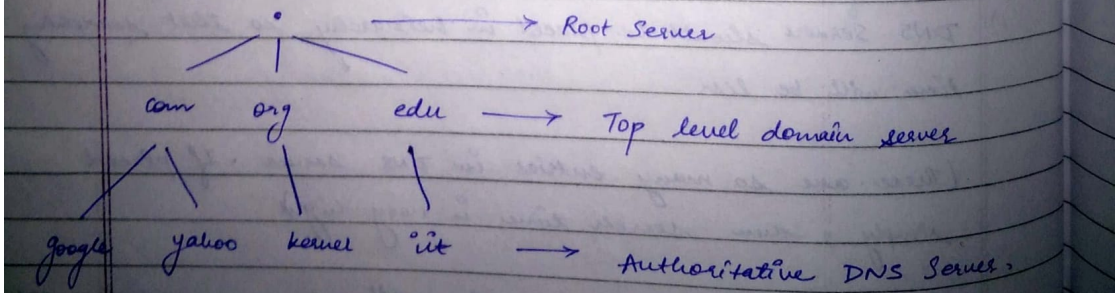
⋮

RFC Standard says that in a dns query we can have maximum 127 levels.

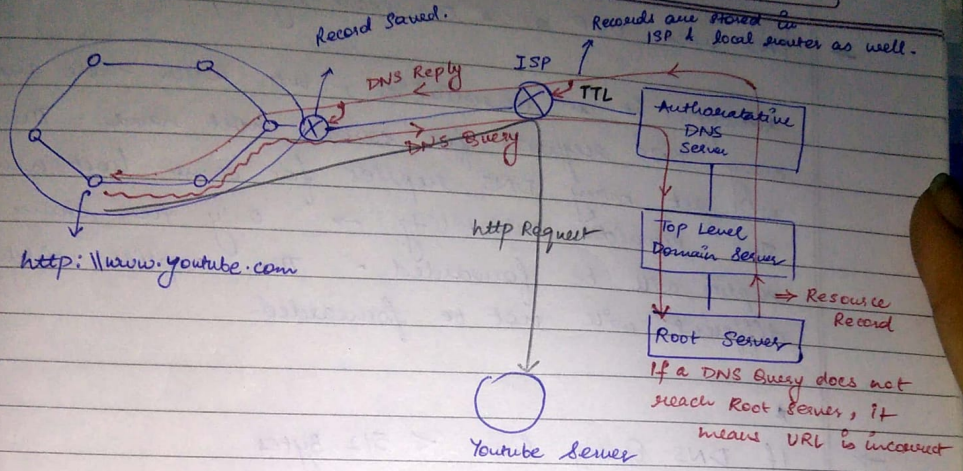
dns query = www . google . com no. of levels = 3

= www . kernel . org no. of levels = 3

= www . iit . edu



Date : / /
Page No.



The resource record is also saved in local router and in ISP

∴ They are called local ~~level~~ DNS Server
or Cache DNS Server
or Proxy DNS Server.

ISP & local router ~~are~~ have small memory & ~~cannot~~ store all the entries

∴ TTL is used as a timer for records.

If TTL ≥ 24 hrs ⇒ Stable record
TTL < 24 hrs ⇒ Unstable record.

Which protocol should be used for transport layer protocol?

→ UDP — It is fast
but is unreliable

→ TCP — It is reliable
but it is slow.

Date : / /
Page No.

→ DNS uses UDP as a transport layer protocol

Although it is unreliable, but when there are many DNS requests for same host name. Then ISP will get many DNS replies for same host name. This provides reliability → only the same DNS replies will be forwarded. The replies which are different will not be forwarded.

*

→ If DNS Query Size < 512 Bytes

then UDP is used as T.L.

→ If DNS Query Size > 512 Bytes

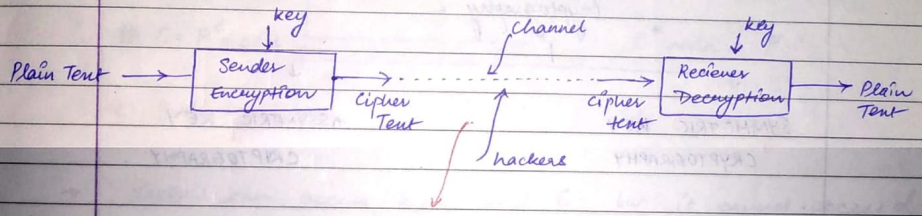
then TCP is used as T.L.

03/06/18

CLASSTIME | Page No.
Date 03/06/18

Security :

Cryptography is a science or art of converting one form of data into other form for providing security to the data.



1.) Cryptography
2.) steganography

Hiding the data behind an image is known as steganography

Cryptogra — Key

- Public key
- Private key

Whenever a key is transmitted over channel and later used for encryption / decryption is known as public key

If the key is kept as secret & later used for encryption & decryption it is known as private key

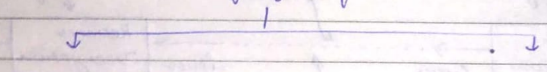
CLASSTIME | Page No.
Date / /

Features of Cryptography :

1.) CONFIDENTIALITY : Providing secrecy to the data is known as confidentiality.

2.) AUTHORIZATION : Proving user's identity & integrity of (AUTHENTICATION) the user is known as authorization
   a) Authentication of user
   b) Authentication of data.

Cryptography :



SYMMETRIC KEY CRYPTOGRAPHY	ASSYMETRIC KEY CRYPTOGRAPHY
→ same key is used for encryption & decryption	→ different keys are used for encryption & decryption.
Ex: Diffie Hellman key exchange algorithm.	Ex: RSA Algorithm

Key features of Cryptography : (We use anything which cannot be guessed)

1.) Prime Numbers : They are not easily guessed as they are not divisible by any no.

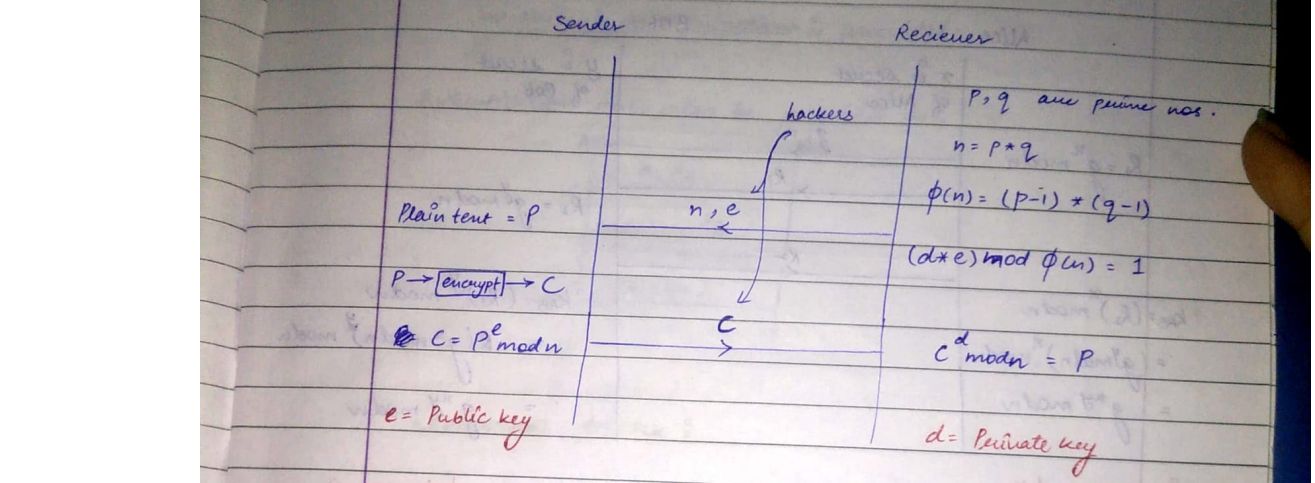
2.) Random Numbers : We can use random no. like OTP because it cannot be guessed easily.

3.) key : → Public key
          → Private key

4.) Time Stamp.

CLASSTIME | Page No.
Date / /

RSA Algorithm:



→ Hackers can access n, e and C but it cannot access d
∴ Hacker cannot convert Cipher text, C to plain text, P
So, Confidentiality is provided.

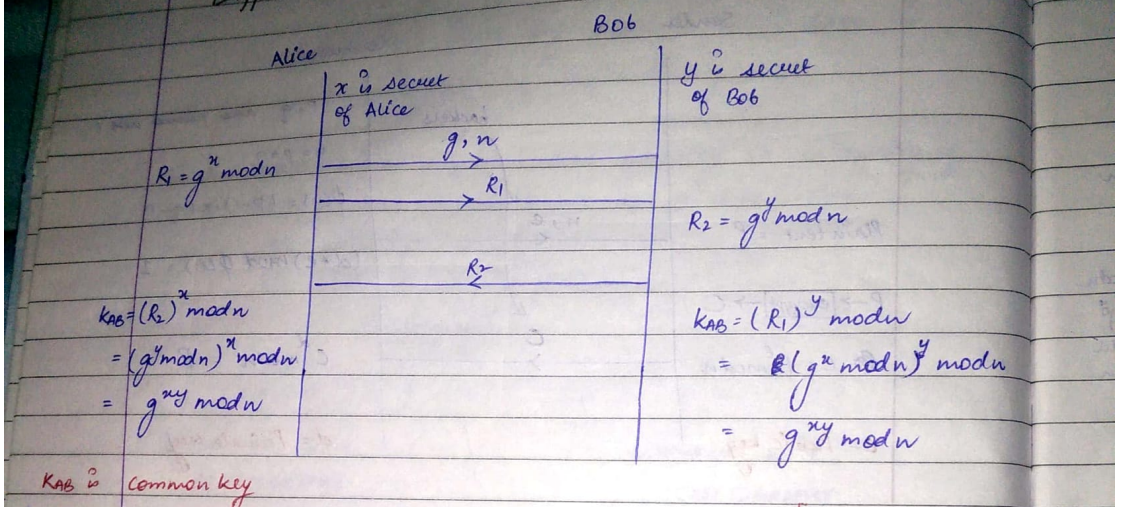
→ At senders side, plain text is encrypted to cipher text using public key of reciever

At reciever's side, cipher text is decrypted to plain text using its own private key or private key of reciever

→ RSA Algorithm is known as asymmetric key cryptography or public key cryptography because ~~sender & reciever~~ encryption & decryption are done by different keys.

CLASSTIME | Page No.
Date / /

Diffie Hellmann key Exchange :

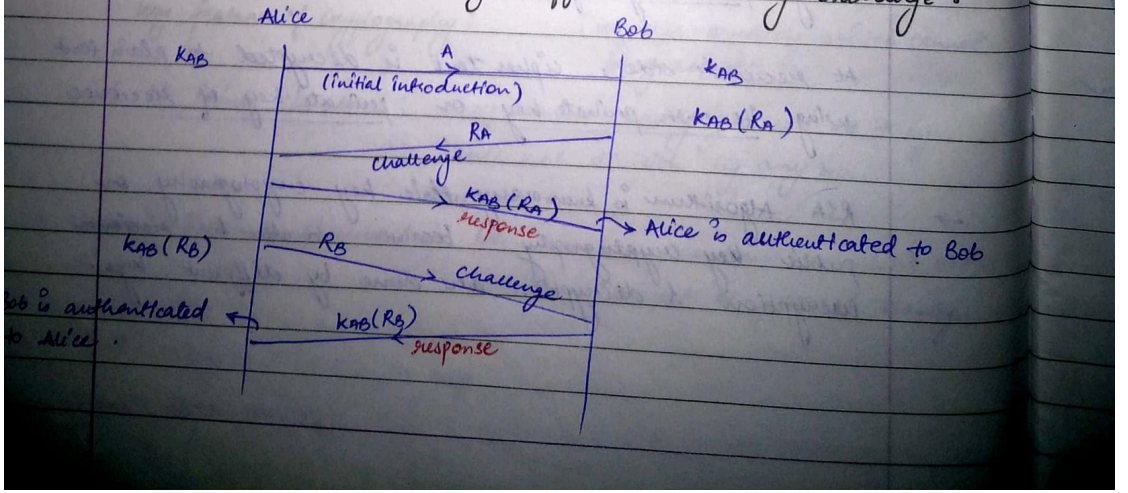


K_{AB} is common key

[$4 \bmod 5 = 4$
$4 \bmod 5 \bmod 5 = 4$
$\therefore = 4 \bmod 5$]

→ Diffie Hellmann key exchange algo is symmetric key cryptography

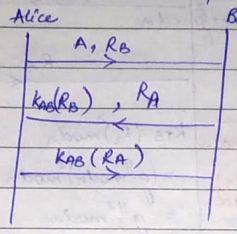
Mutual Authentication using Diffie Hellman key exchange :



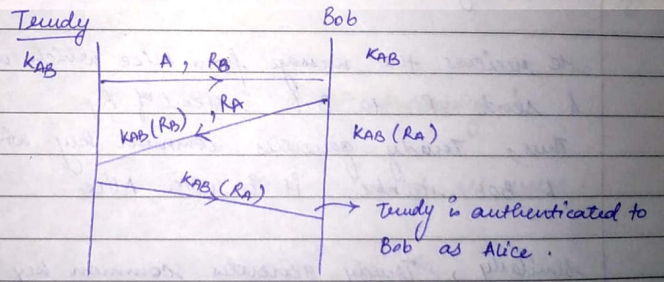
→ Diffie Hellmann key exchange is used for authentication.

→ Mutual Authorization is provided.

Authentication can also be provided as:



REFLECTION ATTACK:



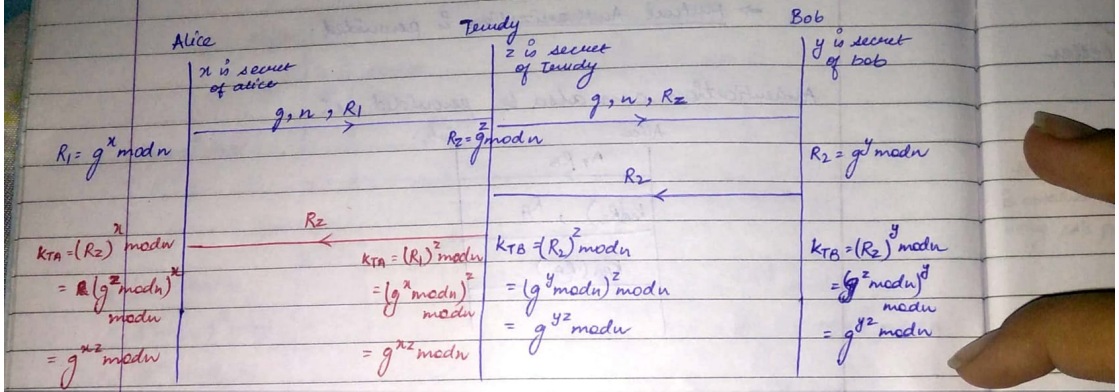
Thus, authentication using diffie Hellmann key can be broken using reflection attack.

So, authentication is failed because bob wrongly authenticate Teudy thinking that it is right.

→ Teudy gets KAB using man in the middle attack.

CLASSTIME | Page No.
Date / /

MAN IN THE MIDDLE ATTACK :



Here, Teendy is the man in the middle.

He recieves the message from Alice which was destined to Bob & sends R_2 to Bob instead of R_1

Thus, Teendy generates common key with Bob wrongly & Bob thinks it is with Alice.

Similarly, Teendy generates common key with Alice wrongly

Both Alice & Bob are in elusion that they have common key with each other, but only hacker has common key with both.

This is the man in middle attack.

CLASSTIME | Page No.
Date / /

→ Drawback: In diffie Hellmann key exchange, if a person x wants to communicate with n people, then n keys are required.

(X) k_{xy} (Y) k_{xy}

(X) k_{xz} (Z) k_{xz}

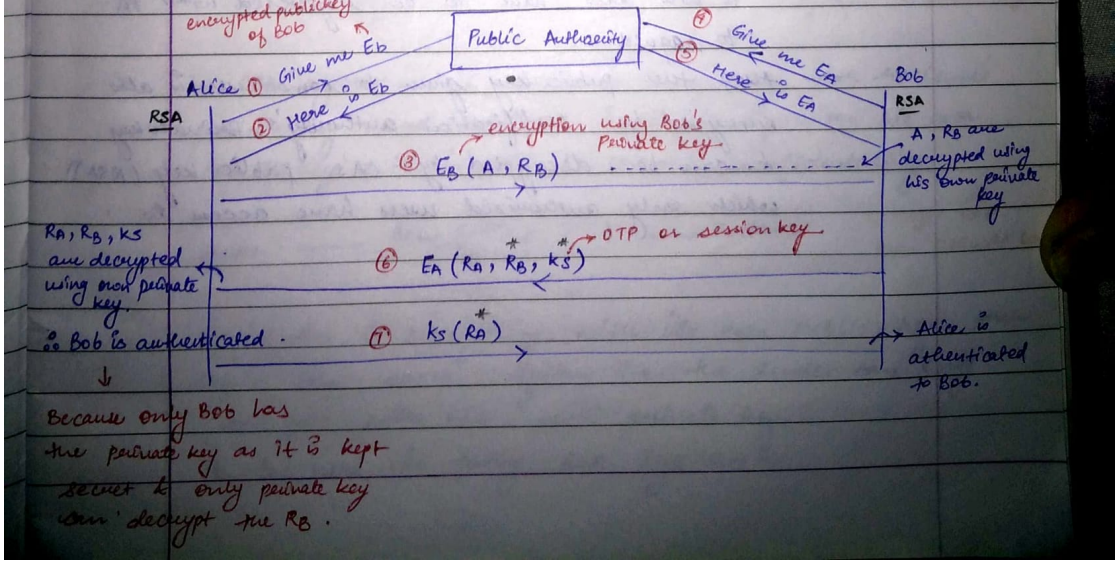
(X) k_{xw} (W) k_{xw}

- Remembering all the key values for maintaining the database of all the key values is difficult.

So, burden of the key value is taken care by third party or public authority.

Mutual Authentication Using Public Authority:

Public Authority contains public keys of all users.



CLASSTIME | Page No.
Date / /

→ Mutual Authentication using RSA algorithm is better than mutual Auth. using DHK in terms of security

→ Mutual authentication using deffie hellmann key is better than mutual auth. using RSA in terms of speed.

→ Public authority is a database which stores the public keys of all the users.

→ Public ~~key~~ authority is also a certification authority.

⇒ Public authority does not store public key of any and every user. First the user will be authorized by PA.

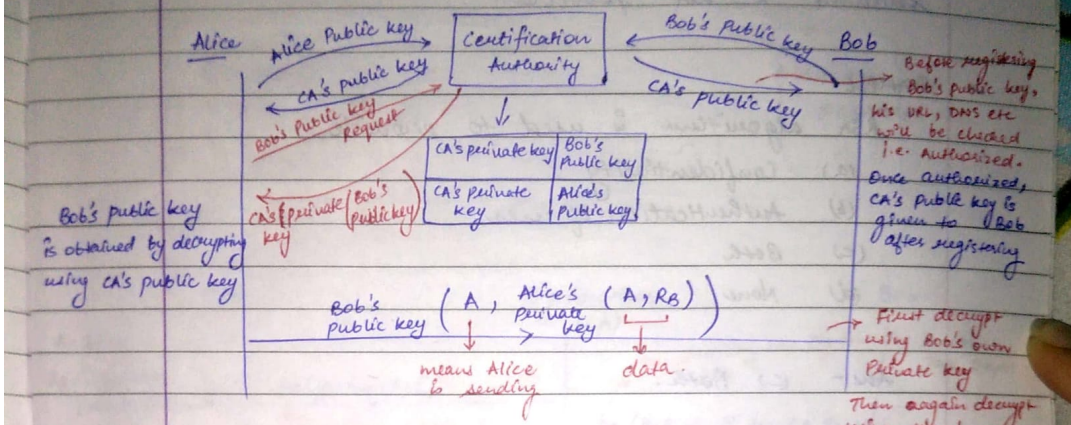
⇒ Also, it does not send the public key of a user to anyone requesting for it. It will also have to be registered with PA to request.

⇒ Also, the public key given to some user is also encrypted by certification authority's private key & can be decrypted by CA's public key (RSA) which only authorized users have access to.

Bob's
is obt
using

CLASSTIME | Page No.
Date / /

CERTIFICATION AUTHORITY



→ In RSA algorithm, if sender is encrypting with his private key & reciever is decrypting ~~in~~ sender's public key, it is used to provide authentication.

Sender → Alice's private key (A, RB) → Reciever
Decrypt using Alice's public key

→ In RSA algorithm, if sender is encrypting with reciever's public key and reciever is decrypting with his own private key, it is used to provide confidentiality

Sender → Bob's public key ( ) → Reciever
Decrypt with own private key

→ If sender is encrypting with its own public key, then again the same system i.e. the sender can only decrypt with his own private key.
It is used for testing RSA algorithm.

CLASSTIME | Page No.
Date / /

→ If sender is encrypting with receiver's private key, this statement is not possible.

→ EXAMPLE:

RSA algorithm is used to provide

(a) Confidentiality
(b) Authentication of user
(c) Both
(d) None

Ans - (c) Both.

AUTHENTICATION OF DATA:

In network security, authentication for data can be provided using digital signatures.

In case of hand written signature, both data & signature cannot be seperated whereas in digital signatures, both data & digital signatures can be seperated.

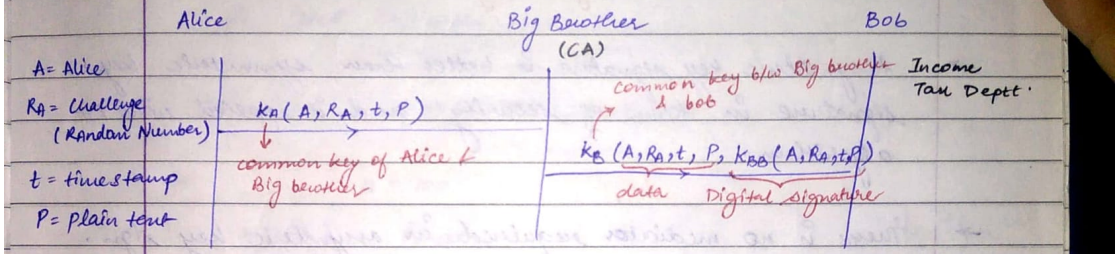
For all types of data, handwritten signature is same whereas in network, forevery individual data, a seperate digital signature is created.

CLASSTIME | Page No.
Date / /

DIGITAL SIGNATURE

- SYMMETRIC KEY SIGNATURE
- ASSYMETRIC KEY SIGNATURE

1.) SYMMETRIC KEY SIGNATURE :



→ In symmetric key signature, the entire algorithm is based on big brother, so it will not work correctly.

Let Alice gives his tax files to CA i.e. Big Brother who acts as a mediator and verifies the files → adds the digital signature & forwards the file along with DG to income tax deptt.

If income tax deptt gets to know that files of Alice are false
- DG can save A by proving that it was mistake by his end (ex. computer)

OR

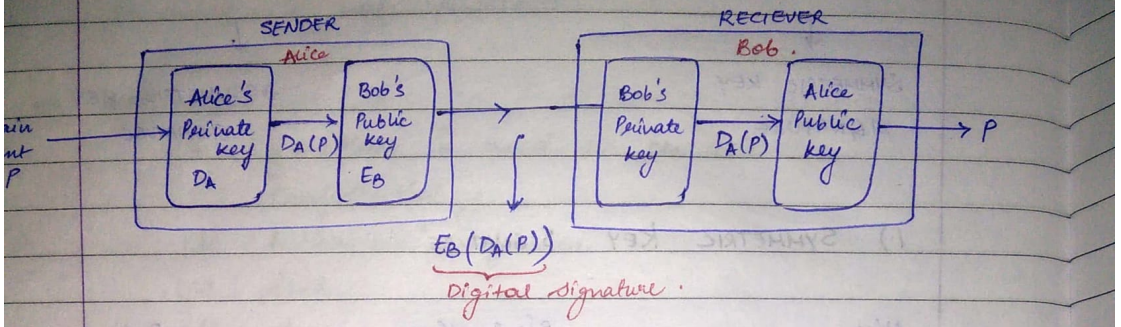
- DG can be honest & show that digital signature generated matches the file ∴ files are falsely sent by A

∴ Symmetric key sign depends on mediator & is lesser secure.

→ Symmetric key signature is better than assymteric in terms of speed and uses diffie-hellman key.

CLASSTIME | Page No.
Date / /

2. ASSYMTRIC KEY SIGNATURE :



→ Assymtric key signature is better than symmetric key signature in terms of security and is applied with RSA algorithm.

→ There is no mediator required in assymtric key sign. and the data is directly encrypted in digital signature.

On decrypting the sign. we get the data.

Q Can we swap both encrypting blocks ?

i.e. Bob's Public key → Alice Private key → on sender side

In this case, digital signature becomes $D_A(E_B(P))$

D_A is public key of Alice & it is shared over the channel decrypted by

Thus, the digital signature becomes less secure. but it will still work.

So, we should not swap encryptions.

Assymetric key signature can be made fastes algo by using message digest :

MESSAGE DIGEST :

Message digest converts a large size message to small size Now encrypting the small message will requires lesser time

∴ Speed increases.

Plain Text , P = (14150789106101051)

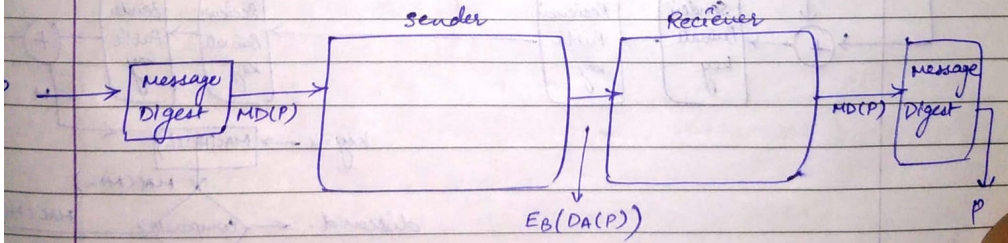
n = 100

MD(P) = P mod n

(message digest)

= (14150789106101051) mod 100

MD(P) = 51



→ For Given P, anyone can calculate MD(P) easily

but for given MD(P), no one can easily calculate P such that MD(P) = = MD(P') & P' = = P

[Ex: MD(P) = 51 P = 151
= 251
= 351]

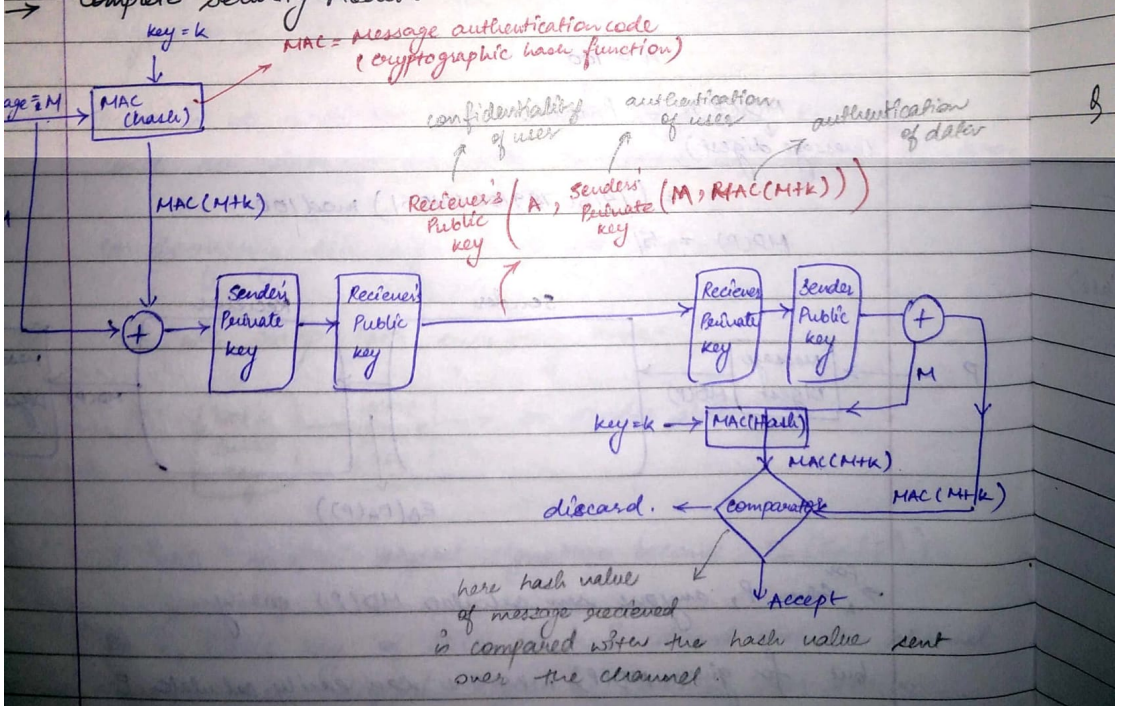
→ Thus, security also increases by using message digest along with speed.

→ The plain text, P can be regenerated at reciever only by the same message digest.

→ SHA → secure hashing algorithm is used as a hash function in message digest.

(MD4 & MD5 are no longer used)

→ Complete security Model:



→ Both RSA and diffie-hellman key algorithms are used (DHK is used for the key in generation of MAC on both sender & reciever side)

→ using birthday attack or paradox attack, sender can modify the data and get it accepted by the receiver, if it knows the hash value.

If manager at sender side knows that data is 151 & its hash will be 51, then he can send some other data with same hash (eg=251) ∴ message = (251, 51). When it reaches receiver, hash generated will be 51 & hash received is also 51 ∴ Message is wrongly accepted

Q In the key generation of RSA algorithm, how many systems are involved?

Here, p & q are two prime nos. chosen by one system only. That system is sufficient to generate the key.
∴ Only one system is involved.

Q In the diffie hellman key algorithm, in the key generation, how many systems are involved?

Both the systems have their own secret x & y which together can create a common key
∴ Two systems are involved.

CLASSTIME | Page No.
Date / /

IPv6 Protocol :

Packet = Base Header | Data

IPv6 has 40 bytes fixed header

40 bytes → ← 65535 bytes

VER = 0100 = IPv4
0110 = IPv6

VER 4 bits	PRIORITY CLASS 8 bits	FLOW LABEL 20 bytes.
PAYLOAD LENGTH 16 bits	NEXT HEADER 8 bits	HOP LIMIT 8 bits
S. IP 128 bits		
D. IP 128 bits		

→ 2^{20} possible flow labels.

Each row = 32 bits
∴ 4 rows to store 128 bits ↓ IPv6 address

→ FTP packet has higher priority than SMTP Packet

(because to download the file user must be online ∴ Router must hurry to send the packet.
whereas a mail can be sent even when user is not available)

→ Foreground packet has higher priority than background packet

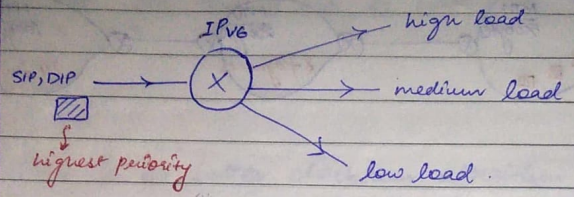
www.cricinfo.com

foreground packet → Live Cricket Score | background packet

→ Out of all packets, LSP has highest priority.

CLASSTIME | Page No.
Date / /

→ RSVP & RTP :

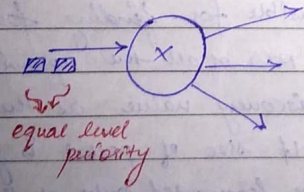


For the highest priority packet, it must go through the low load path
& for a lowest priority packet, it should be forwarded to high load path.

* This selection of path till the destination is achieved by RSVP & RTP

RSVP → Resource reservation protocol
RTP → Real time protocol.

→ Flow Label :

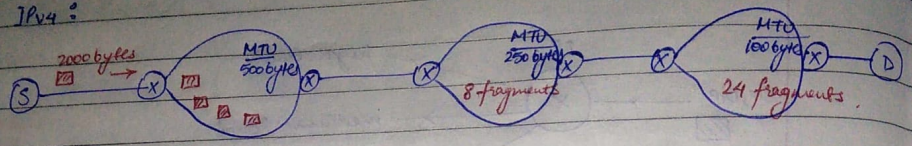


If both the packets have same level of priority, they can still be distinguished based on flow label.

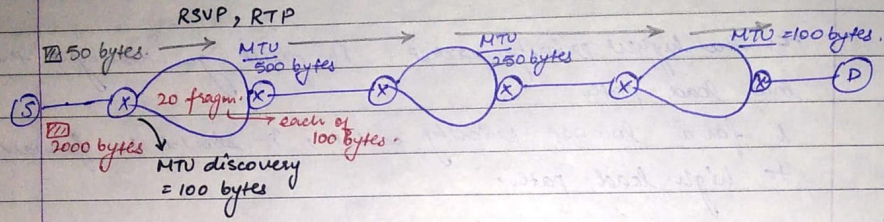
→ The fragmentation options has been removed from IPv6 which was present in IPv4 → DF, MF, Id, fragmentation offset etc.

Date

IPv4 :



IPv6 :



→ In IPv4, fragmentation is compulsory. so it is placed in main header
whereas in IPv6, fragmentation is an option i.e. if packet size is more than MTU discovery, then only packet is fragmented.

→ In IPv6, RSVP & RTP are responsible for finding forwarding path. so RSVP & RTP knows about MTU of all the upcoming routers. ∴ It calculates MTU discovery value as the min. MTU value on the complete path. If size of packet is greater than this MTU value, then it will be fragmented only at the beginning. After that it will be only forwarded. ∴ Increases speed as not again & again check & fragmentation.

→ Payload Length in IPv6 indicates the data in the packet

→ Next header in IPv6 indicates whether any other headers are added along with base header
If fragments are present, then it stores the address of next packet header. If no fragments, then stores null.

CLASSTIME | Page No.
Date / /

* → The purpose of hop limit is to identify if any loop will exist for the packet or not.

→ There is no better checksum algorithm than IPv6.