

Unit-5 Notes- Transport Layer

DESIGN ISSUES OF TRANSPORT LAYER:-

Transport layer is responsible for following issues:-

- a) Accepting message segments from the application layer and to divide into packets.
- b) End-to-End Delivery of the packet
- c) Combining packets into message segment at receiver side.
- d) Connection management.

In other words transport layer is responsible for two tasks:-

- ▶ Transport and regulate the flow of information from source to destination, reliably and accurately.
- ▶ The end-to-end control:
 - ▶ Sliding windows.
 - ▶ Sequencing numbers.
 - ▶ Acknowledgments.
 - ▶ Segmentation.
 - ▶ Multiplexing.

DESIGN ISSUES OF SESSION LAYER:-

The session layer is level five of the seven level OSI model. It responds to service requests from the presentation layer and issues service requests to the transport layer. the session layer provides the mechanism for opening, closing and managing a session between end-user application processes, i.e. a semi-permanent dialogue. Communication sessions consist of requests and responses that occur between applications. Session layers are commonly used in application environments that make use of remote procedure calls (RPCs).

An example of a session layer protocol X.225 or ISO 8327. In case of a connection loss this protocol may try to recover the connection. If a connection is not used for a long period, the session layer protocol may close it and re-open it. It provides for either full duplex or half-duplex operation and provides synchronization points in the stream of exchanged messages. List of Session layer services:-

- **Authentication**
- **Permissions**
- **Session restoration (checkpointng and recovery)**

Authentication is the act of establishing or confirming something (or someone) as authentic, that is, that claims made by or about the thing are true. This might involve confirming the identity of a person, the origins of an artifact, or assuring that a computer program is a trusted one.

Permissions or Access control

One familiar use of authentication and authorization is access control. A computer system supposed to be used only by those authorized must attempt to detect and exclude the unauthorized. Access to it is therefore usually controlled by insisting on an authentication procedure to establish with some established degree of confidence the identity of the user, thence granting those privileges as may be authorized to that identity.

In some cases, ease of access is balanced against the strictness of access checks. For example, the credit card network does not require a personal identification number, and small transactions usually do not even require a signature. The security of the system is maintained by limiting distribution of credit card numbers, and by the threat of punishment for fraud.

Checkpoints

Session layer is responsible for creating several checkpoints, checkpoints are also treated as recovery points i.e. in case of failure the system rollback to its previous checkpoint configuration or action.

REMOTE PROCEDURE CALL (RPC): -

Remote Procedure Call (RPC) provides a different paradigm for accessing network services. Instead of accessing remote services by sending and receiving messages, a client invokes services by making a local procedure call. The local procedure hides the details of the network communication. RPC is a powerful technique for constructing distributed, client-server based applications. It is based on extending the notion of conventional, or local procedure calling, so that the called procedure need not exist in the same address space as the calling procedure. The two processes may be on the same system, or they may be on different systems with a network connecting them. By using RPC, programmers of distributed applications avoid the details of the interface with the network. The transport independence of RPC isolates the application from the physical and logical elements of the data communications mechanism and allows the application to use a variety of transports. There are 3 components on each side in case of RPC System

- a user program (client or server)
- a set of stub procedures
- RPC runtime support

Server program defines the server's interface using an interface definition language (IDL).The IDL specifies the names, parameters, and types for all client-callable server procedures.

A stub compiler reads the IDL and produces two stub procedures for each server procedure: a client-side stub and a server-side stub

The server writer writes the server and links it with the server-side stubs; the client writes her program and links it with the client-side stubs.

The stubs are responsible for managing all details of the remote communication between client and server.

A client-side stub is a procedure that looks to the client as if it were a callable server procedure.

- A server-side stub looks to the server as if it's a calling client.
- The client program thinks it is calling the server; in fact, it's calling the client stub.
- The server program thinks it's called by the client; in fact, it's called by the server stub.
- The stubs send messages to each other to make the RPC happen.

Binding:-

- Binding is the process of connecting the client and server
- The server, when it starts up, exports its interface, identifying itself to a network name server and telling the local runtime its dispatcher address.
- The client, before issuing any calls, imports the server, which causes the RPC runtime to lookup the server through the name service and contact the requested server to setup a connection.
- The import and export are explicit calls in the code.

Marshalling:

- Marshalling is the packing of procedure parameters into a message packet.
- The RPC stubs call type-specific procedures to marshall (or unmarshall) all of the parameters to the call.
- On the client side, the client stub marshalls the parameters into the call packet; on the server side the server stub unmarshalls the parameters in order to call the server's procedure.
- On the return, the server stub marshalls return parameters into the return packet; the client stub unmarshalls return parameters and returns to the client.

DESIGN ISSUES WITH PRESENTATION LAYER:-

The primary goal of this layer is to take care of the syntax and semantics of the information exchanged between two communicating systems. Presentation layer takes care that the data is sent in such a way that the receiver will understand the information(data) and will be able to use the data Languages (syntax) can be different of the two communicating systems. Under this condition presentation layer plays a role translator.

In order to make it possible for computers with different data representations to communicate, the data structures to be exchanged can be defined in an abstract way. The presentation layer manages these abstract data structures and allows higher-level data structures(eg: banking records), to be defined and exchanged.

- To manage and maintain the Syntax and Semantics of the information transmitted.
- Encoding data in a standard agreed upon way. Eg: String, double, date, etc.
- Perform Standard Encoding on wire.

Functions of Presentation Layer

1. Translation: Before being transmitted, information in the form of characters and numbers should be changed to bit streams. The presentation layer is responsible for interoperability between encoding methods as different computers use different encoding methods. It translates data between the formats the network requires and the format the computer.
2. Encryption: It carries out encryption at the transmitter and decryption at the receiver.
3. Compression: It carries out data compression to reduce the bandwidth of the data to be transmitted. The primary role of Data compression is to reduce the number of bits to be transmitted. It is important in transmitting multimedia such as audio, video, text etc.

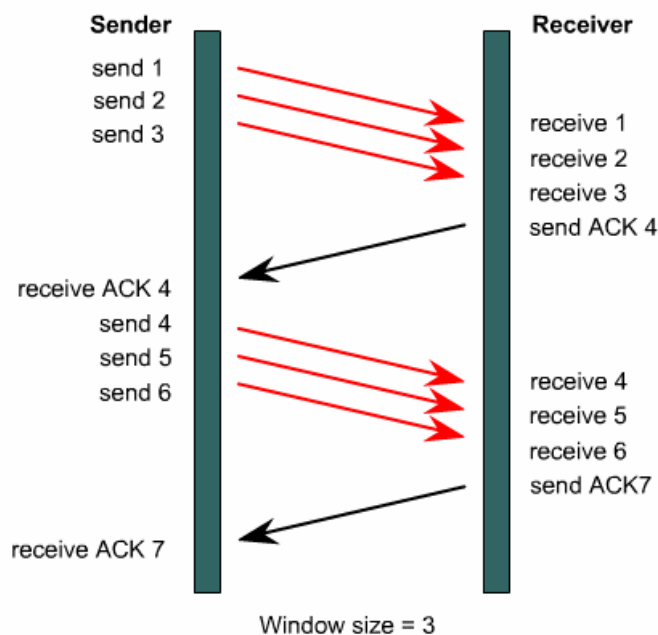
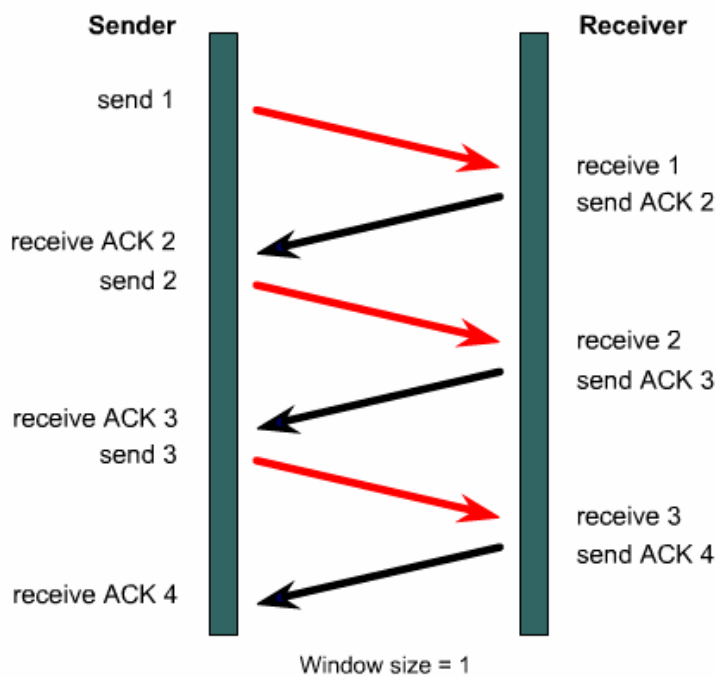
TCP-WINDOW MANAGEMENT SYSTEM:-

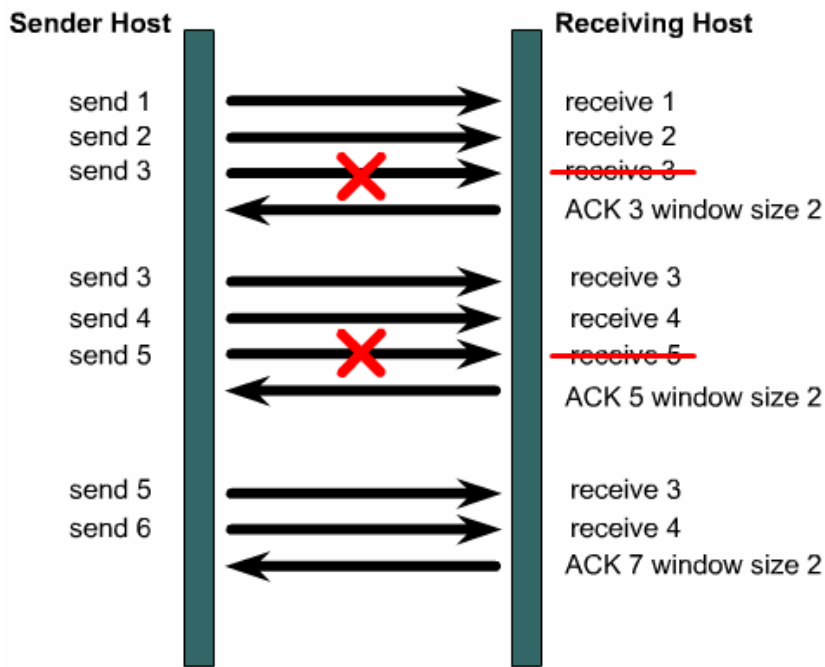
Windowing and Window Size: Window management in TCP is an important concept that ensures reliability in packet delivery as well as reduce the wastage of time in waiting for the acknowledge after each packet.

Window size: window size determines the amount of data that you can transmit before receiving an acknowledgment. Sliding window refers to the fact that the window size is negotiated dynamically during the TCP session.

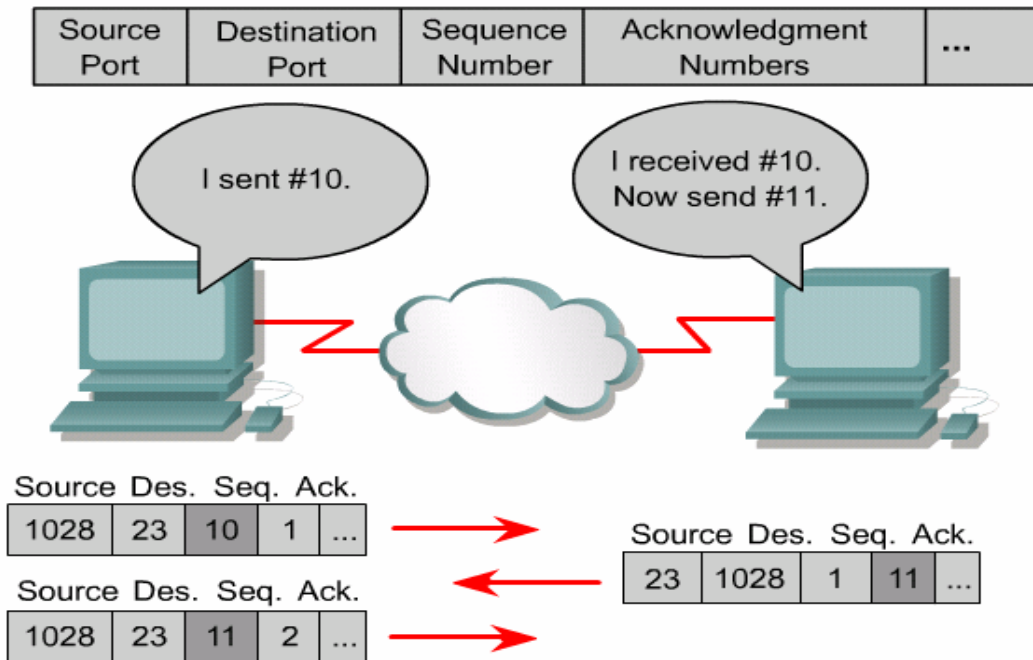
1. Expectational acknowledgment means that the acknowledgment number refers to the octet that is next expected
2. If the source receives no acknowledgment, it knows to retransmit at a slower rate.

The mechanism of the sliding window style may be understood easily with the help of below given diagrams:





TCP Sequence and Acknowledgement



DATA COMPRESSION TECHNIQUES:-

डेटा कम्प्रेसन एक ऐसी प्रक्रिया है जिसमें डेटा बिट्स को कम कर दिया जाता है जिससे कि डेटा का आकार कम हो जाता है तथा इससे स्टोरेज क्षमता तथा फ़ाइल ट्रांसफर की गति बढ़ जाती है। डेटा कम्प्रेसन का सबसे बड़ा लाभ यह है कि हम समान प्रकार के डेटा को कम बिट्स में सुरक्षित रख सकते हैं और संचारित कर सकते हैं। डेटा को कम्प्रेस करने के लिए फार्मूला तथा एल्गोरिथ्म का प्रयोग किया जाता है। उदाहरण के तौर पर मान लीजिये जब हम किसी भी गाने को डाउनलोड करते हैं तो वही गाना अलग-अलग साइज़ का होता है (10mb, 5mb, 2.6mb) जिससे कि हम तेज गति से उसे डाउनलोड कर सकते हैं।

कम्प्रेसन तकनीक के प्रकार

डेटा कम्प्रेसन दो प्रकार का होता है जो निम्न हैं:-

1:-Lossless कम्प्रेसन:- वह कम्प्रेसन जिसमें डेटा के आकार को इस तरह घटाया जाता है जिससे कि डेटा की हानि नहीं होती है Lossless कम्प्रेसन कहलाता है। इसका मतलब ये हुआ कि हम decompressed करके वापस डेटा के मूल साइज़ को वापस पा सकते हैं। इस कम्प्रेसन का प्रयोग ऐसे डाक्यूमेंट्स में किया जाता है जो टेक्स्ट को रखे रहते हैं। loss less compression एक प्रकार का data compression होता है जो कि 2:1 के ratio में data को compress करता है जब कोई भी data को loss less data compression algorithm के द्वारा compress होता है और फिर जब उसे decompress किया जाता है तो वह पूरी तरह से original data recover होता है। यह data compression text और data files को compress करने के लिए उपयोग किया जाता है जैसे कि bank records , Whats App text message , text articles etc. इस data compression को text और data files को Compress करने के लिए इसलिए use किया जाता है क्योंकि यह data compress technique data को decompress करते time original data के form में ला देता है मतलब की इस technique से कोई data loss नहीं करता है

2:-Lossy कम्प्रेसन:- वह कम्प्रेसन जिसमें डेटा की हानि होती है Lossy कम्प्रेसन कहलाता है तथा हम दुबारा decompressed करके उसको वापस नहीं पा सकते हैं इसका मत्लाब ये हुआ कि इसमें डेटा हमेशा के लिए delete हो जाता है। वह compression जिसमें data का loss होता है उसे lossy compression कहते हैं। जिसमें हम data को decompressed करते time पर कुछ ना कुछ data delete हो जाता है Lossy compression एक high compression ration होता है जो कि किसी भी data को 50:1 के form में compress करता है lossy compression में data को compress करने के बाद जब decompress करते हैं मतलब दुबारा original form में लाते हैं तो कुछ data loss हो जाता है मतलब की completely original data recover नहीं हो पाता है Lossy data compression Audio , video और image को compress करने के लिए use किया जाता है lossy algorithm lossless algorithm के अपेक्षा बहुत की अच्छा compression ration होता है क्योंकि यह data के size को ज्यादा compress कर देता है। इस कम्प्रेसन का प्रयोग ज्यादातर चित्र, ऑडियो एवं ग्राफिक्स में किया जाता है।

कुछ डाटा कम्प्रेसन तकनीक

कुछ बेसिक डाटा कम्प्रेसन तकनीक का जिक्र नीचे किया गया है जो अक्सर प्रयोग में लाये जाते हैं:

- **Null कम्प्रेसन:** ये ब्लांक स्पेस कि सीरीज को एक कम्प्रेसन कोड के साथ रिप्लेस कर देता है और उसके बाद एक ऐसी संख्या डालता जो बताते हैं कि उन स्पेस कि संख्या क्या है।
- **Run-length कम्प्रेसन:** ये चार या उस से ज्यादा नल करैक्टर के सीरीज को कॉम्प्रेस कर देता है। इसमें उन चरक्टेर्स को एक कम्प्रेसन कोड से रिप्लेस किया जाता है और फिर उसके बाद वो मान डाला जाता है जो ये बताते हैं कि कितनी बार वो करैक्टर रिपीट हुआ।
- **कीवर्ड इनकोडिंग:** ये एक ऐसा टेबल बनता है जिसमे वो मान होते हैं जो च्चाराक्टेर्स के कॉमन सेट को रेप्रेसेंट करते हैं। उदाहरण के तौर पर मान लीजिये कि *The* और *He* जैसे कैरेक्टर जो बार-बार आते हैं। इनकी जगह पर ऐसे टोकन डाल दिए जाते हैं जो कैरेक्टर को ट्रांसमिट करने के प्रयोग में आते हैं।
- **Adaptive Human Encoding और Lempel Ziv Algorithms:** ये अल्गोरिथम बार-बार आने वाले कैरेक्टर्स कि जगह पर एक सिंबल डिक्शनरी का प्रयोग करते हैं। जैसे जैसे नये पैटर्न आते हैं ये डिक्शनरी अपडेट होती चली जाती है। डाटा के संचार के लिए रिसीविंग सिस्टम को ये डिक्शनरी दे दी जाती है जिस से उन्हें ये पता चल जाता है कि इन कैरेक्टर्स को डिक्कोड कैसे करना है। फाइल स्टोरेज के समय इस डिक्शनरी को कम्प्रेसन फाइल के साथ ही सुरक्षित रखा जाता है।

LOSS COMPRESSION	LOSSLESS COMPRESSION
इस तकनीक में image compression के बाद कुछ नुकसान की जानकारी शामिल है।	जबकि इस तकनीक में image compression के बाद कोई नुकसान शामिल नहीं है।
इस तकनीक का उपयोग करके compress किया गया data ठीक से पुनर्प्राप्त और पुनर्निर्माण नहीं किया जा सकता है।	जबकि इस तकनीक से compress किये गए data को मूल डेटा से पुनर्प्राप्त किया जा सकता है।
यह एप्लिकेशन के लिए उपयोग किया जाता है जो मूल और पुनर्निर्मित डेटा के बीच अंतर सहन कर सकता है।	यह आवेदन के लिए प्रयोग किया जाता है जो मूल और पुनर्निर्मित डेटा के बीच कोई अंतर बर्दाश्त नहीं कर सकता है।
sound और image compression के लिए lossy compression का उपयोग किया जाता है।	जबकि Text compression के लिए lossless compression का उपयोग किया जाता है।
इस चैनल में अधिक डेटा समायोजित किया जा सकता है।	इस चैनल में कम डेटा को समायोजित किया जा सकता है।
इसके उदाहरण हैं (I) Telephone system (ii) Video CD	इसके उदाहरण हैं (i) Fax machine (ii) Radio logical image
इसकी application हैं – JPEG, GUI, MP3, MP4, OGG, H-264, MKV, etc.	इसकी application हैं – RAW, BMP, PNG, WAV, FLAC, ALAC etc.