

A solid red vertical bar is located on the far left side of the image, extending from the top to the bottom.

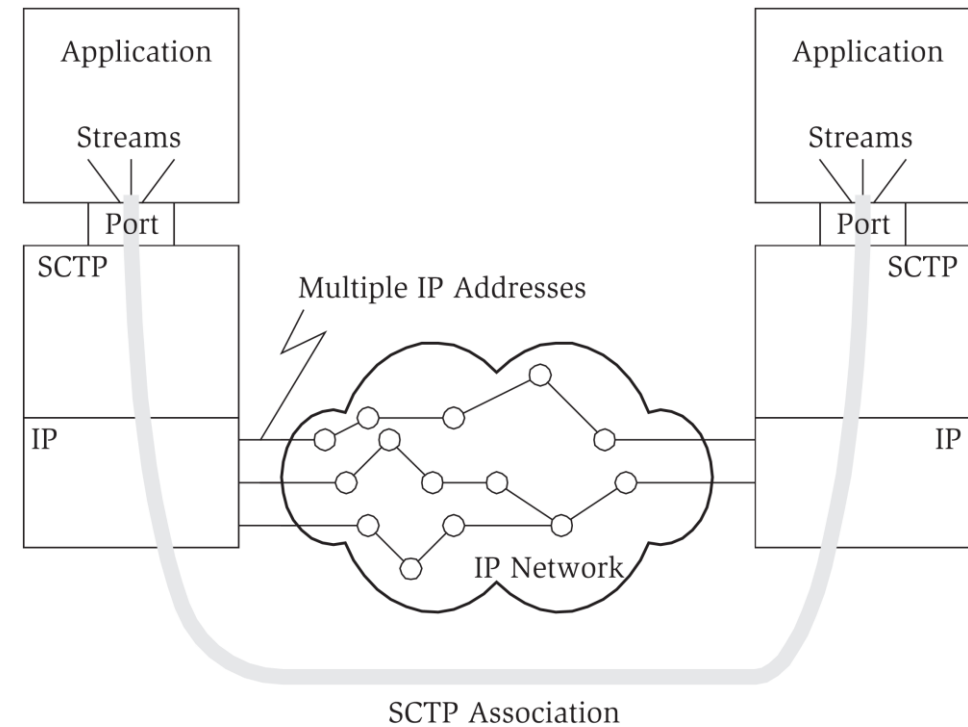
Stream Control Transmission Protocol (SCTP)

Stream Control Transmission Protocol

- Stream Control Transmission Protocol (SCTP) es un nuevo protocolo de la capa de transporte, fiable y orientado a mensaje (RFC 2960).
- SCTP combina las mejores características de UDP y TCP.
- Diseñado para aplicaciones de Internet recientemente introducidas
- Estas aplicaciones nuevas necesitan un servicio más sofisticado que TCP

Servicios SCTP

- Comunicación Proceso a Proceso.
- Múltiples flujos.
- Multihoming.
- Comunicación Full-Duplex.
- Servicio orientado a la conexión.
- Servicio fiable.



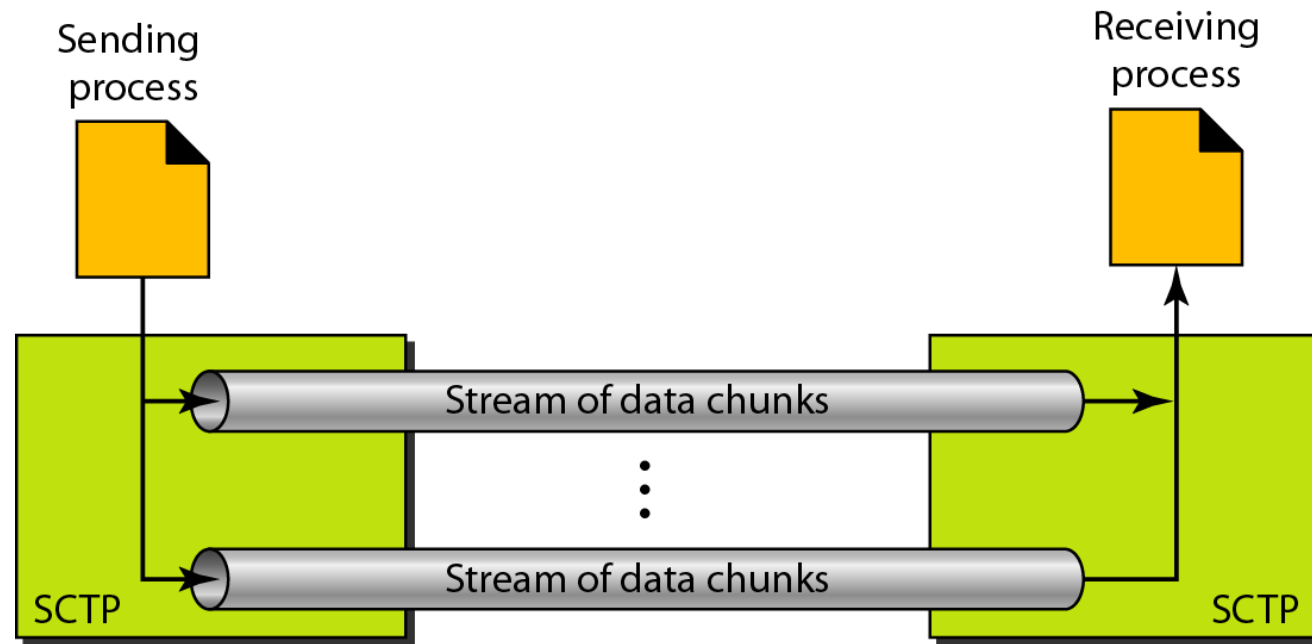
Servicios SCTP: Comunicación Proceso a Proceso

- SCTP usa todos los puertos bien conocidos en el espacio TCP

<i>Protocol</i>	<i>Port Number</i>	<i>Description</i>
IUA	9990	ISDN over IP
M2UA	2904	SS7 telephony signaling
M3UA	2905	SS7 telephony signaling
H.248	2945	Media gateway control
H.323	1718, 1719, 1720, 11720	IP telephony
SIP	5060	IP telephony

Algunos números de puerto extra utilizados por SCTP

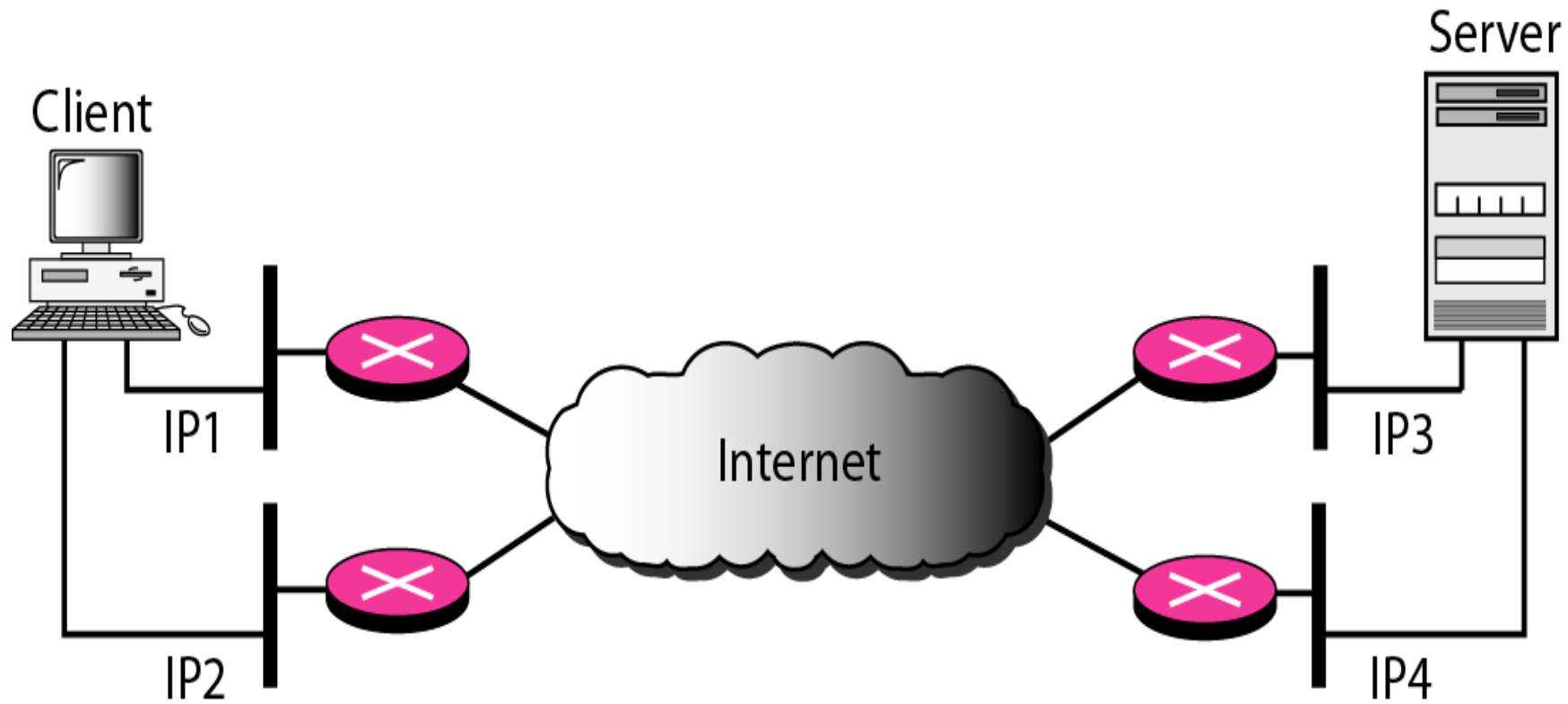
Servicios SCTP: Flujos Múltiples



Servicios SCTP: Flujos Múltiples

- SCTP permite servicio multi flujo en cada conexión.
- Una conexión se denomina asociación en la terminología SCTP.
- SCTP se utiliza principalmente para transferir datos en tiempo real (multimedia).
- Si un flujo se bloquea, los otros flujos aun pueden entregar sus datos (*fault tolerance*).

Servicios SCTP: Multihoming



Servicios SCTP: Multihoming

- El host emisor y receptor puede definir múltiples direcciones IP en cada extremo para una asociación
- Este es un enfoque tolerante a fallas:
 - Cuando un camino falla, se puede utilizar otra interfaz para entregar los datos sin interrupción.
- Una asociación en SCTP puede comprender múltiples flujos

Servicios SCTP: Comunicación Full-Duplex

- SCTP ofrece servicio full-duplex, en el que los datos pueden enviarse en ambas direcciones al mismo tiempo.
- Cada SCTP tiene un buffer de envío y un buffer de recepción.

Servicios SCTP: *Servicio orientado a la conexión*

- SCTP es un protocolo orientado a la conexión.
- Una conexión se denomina una asociación.
- Pasos de conexión:
 - Las dos entidades SCTP establecen una asociación entre ellos.
 - Los datos se intercambian en ambas direcciones
 - La asociación se termina.

Servicios SCTP: *Servicio fiable*

- SCTP, como TCP, es un protocolo de transporte fiable.
- Utiliza un mecanismo de confirmación para verificar la llegada correcta y segura de los datos

Características SCTP

- Número de secuencia de transmisión
- Identificador de flujo
- Número de secuencia de flujo
- Paquetes
- Número de confirmación
- Control de flujo
- Control de errores
- Control de congestionamiento

Características SCTP: *Número de secuencia de transmisión (TSN)*

- La unidad de datos en SCTP es el *data chunk*.
- En SCTP, un data chunk se enumera usando un TSN.
- Similar al número de secuencia en TCP.
- TSN es de 32 bits.
- Se inicializa con un número aleatorio entre 0 y $2^{32}-1$.

Características SCTP: *Identificador de flujo*

- En SCTP puede haber varios flujos en cada asociación
- Cada flujo en SCTP necesita ser identificado utilizando un identificador de flujo (SI).
- Cada data chunk debe portar el SI en su encabezado de forma que cuando este llegue a su destino, este pueda ser propiamente ubicado en su flujo

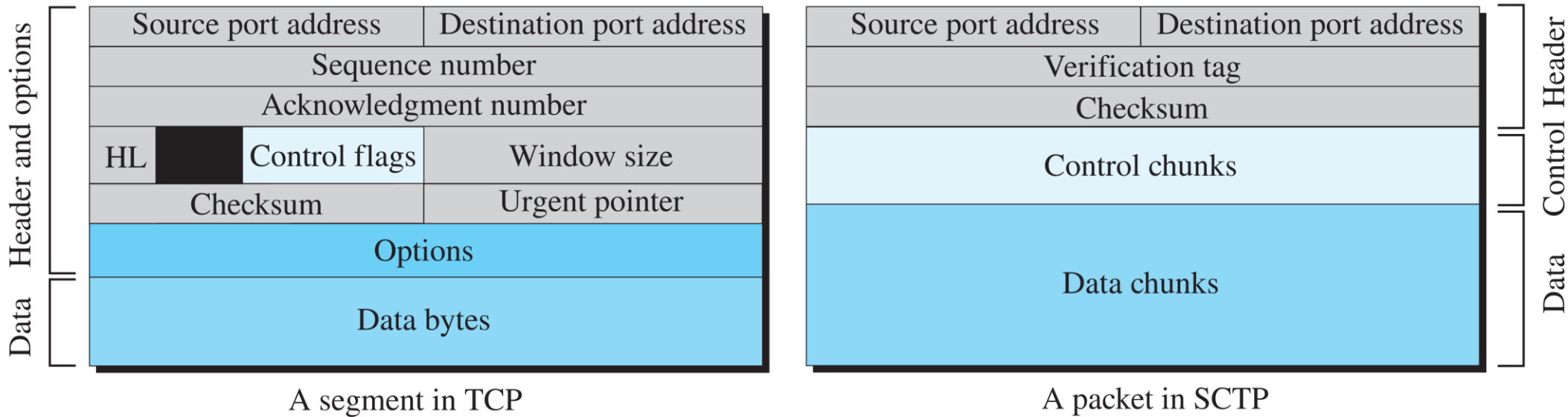
Características SCTP: *Número de secuencia de flujo*

- SCTP usa el número de secuencia de flujo (*SSN*) para diferenciar entre data chunks que pertenecen al mismo flujo
- Esto garantiza que los data chunks de cada flujo son entregados en orden a su destino.

Características SCTP: Paquetes

- Los datos se transportan como *data chunks*.
- La información de control se transporta como *control chunks*.
- Varios *control chunk* y *data chunk* pueden empaquetarse juntos en un paquete.
- Un paquete SCTP juega el mismo rol que un segmento en TCP.

Comparación entre un segmento TCP y un paquete SCTP

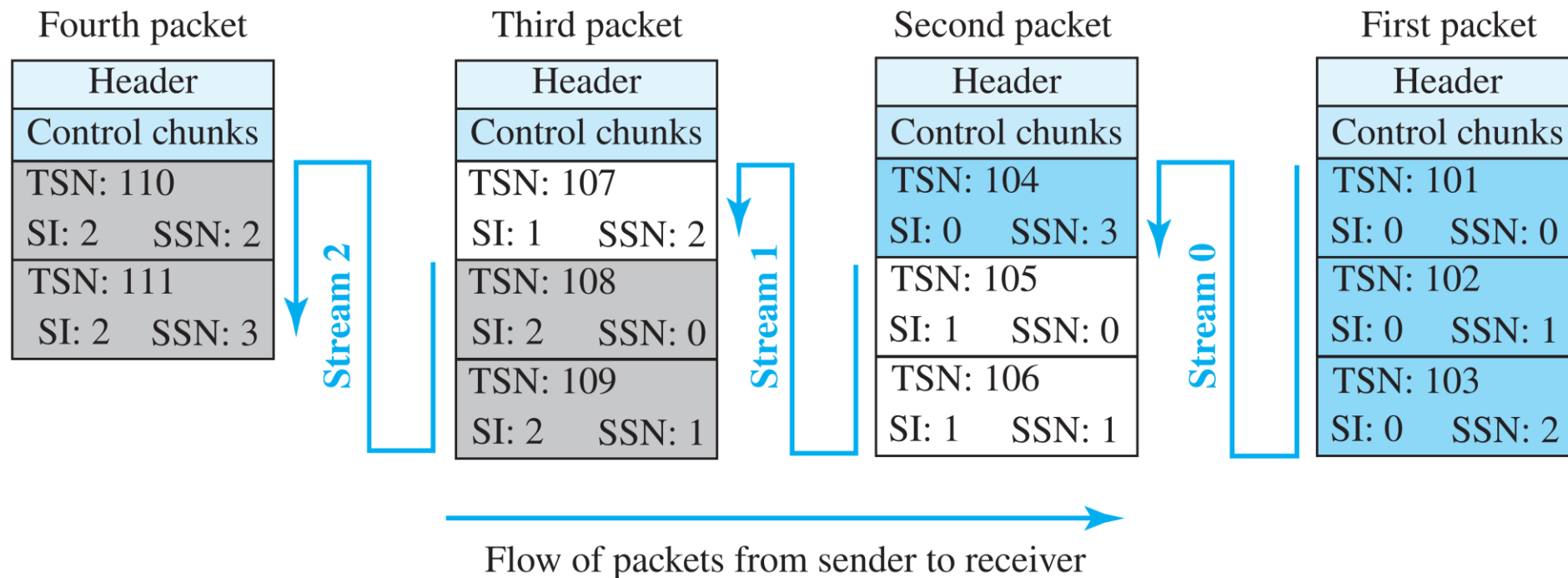


Características SCTP: Paquetes

- En SCTP, la información de control y los datos se transportan en *chunks* separados.
- El encabezado SCTP es más corto por las siguientes razones:
 - Un número de secuencia SCTP (TSN) pertenece a cada *data chunk* y por ende se ubica en el encabezado del *chunk*.
 - El número de confirmación y el tamaño de ventana son parte de cada *control chunk*.
 - No hay necesidad de un campo de longitud de encabezado porque no hay opción de hacer variable la longitud del encabezado; el encabezado SCTP es de longitud fija (12 bytes).
 - No se necesita el campo urgent pointer.

Características SCTP: Paquetes

- Flujo de paquetes SCTP



Características SCTP: Paquetes

- Los *data chunk* se identifican por tres items: TSN, SI, y SSN.
 - TSN es un número acumulativo que identifica la asociación.
 - SI define el flujo.
 - SSN define el *chunk* en un flujo.

Características SCTP: Acknowledgment Number

- Los números de confirmación SCTP son chunk-oriented.
- Los números de confirmación se usan para confirmar solo *data chunks*.
- Los *control chunk* son confirmados por otros *control chunk* de ser necesario.
 - Ejm: el INIT *control chunk* se confirma mediante otro INIT ACK *control chunk*.

Características SCTP: Control de flujo

- SCTP implementa control de flujo para evitar sobrecargar al receptor.

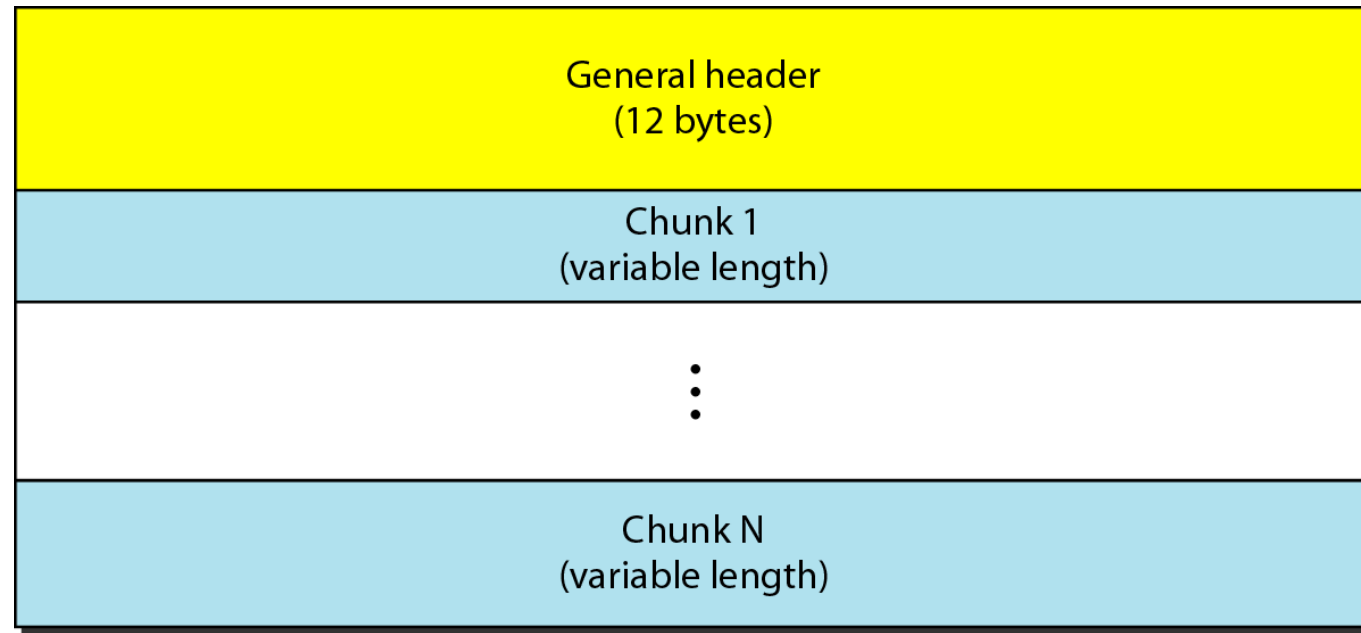
Características SCTP: Control de errores

- SCTP implementa control de errores para proporcionar fiabilidad.
- Se utiliza los números TSN y números de confirmación para controlar errores

Características SCTP: Control de congestionamiento

- SCTP implementa control de congestionamiento para determinar cuantos *data chunk* pueden ser inyectados a la red

Formato de paquete



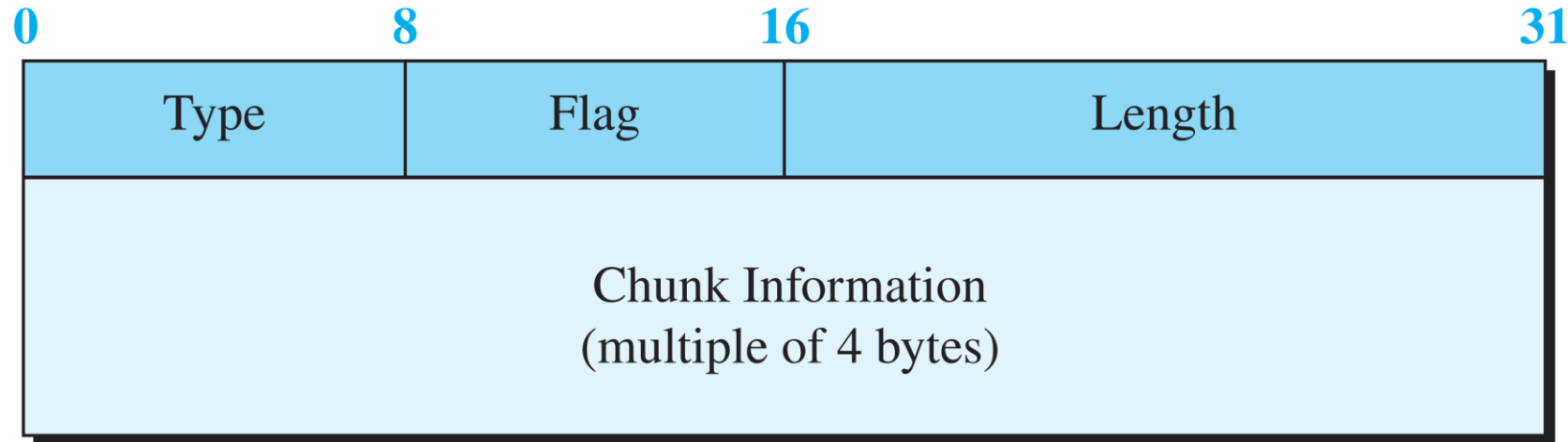
- En un paquete SCTP, los *control chunk* van antes que los *data chunks*.
- Los *control chunks* controlan y mantienen la asociación

Formato de paquete: Encabezado general

Source port address 16 bits	Destination port address 16 bits
Verification tag 32 bits	
Checksum 32 bits	

Formato de paquete: Chunks

- SCTP requiere que la sección de información sea un múltiplo de 4 bytes; de no ser el caso, se agregan bytes de relleno al final de la sección.

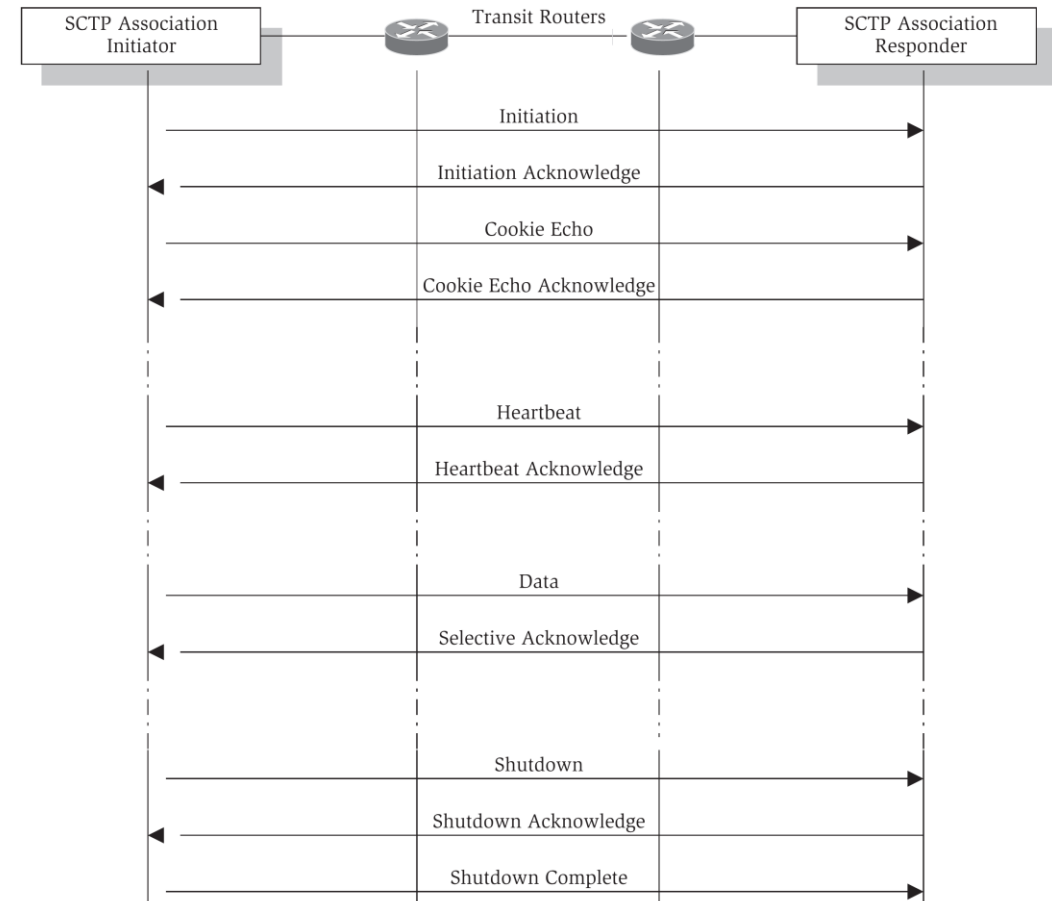


Formato de paquete: Chunks

<i>Type</i>	<i>Chunk</i>	<i>Description</i>
0	DATA	User data
1	INIT	Sets up an association
2	INIT ACK	Acknowledges INIT chunk
3	SACK	Selective acknowledgment
4	HEARTBEAT	Probes the peer for liveliness
5	HEARTBEAT ACK	Acknowledges HEARTBEAT chunk
6	ABORT	Aborts an association
7	SHUTDOWN	Terminates an association
8	SHUTDOWN ACK	Acknowledges SHUTDOWN chunk
9	ERROR	Reports errors without shutting down
10	COOKIE ECHO	Third packet in association establishment
11	COOKIE ACK	Acknowledges COOKIE ECHO chunk
14	SHUTDOWN COMPLETE	Third packet in association termination
192	FORWARD TSN	For adjusting cumulating TSN

Una Asociación SCTP

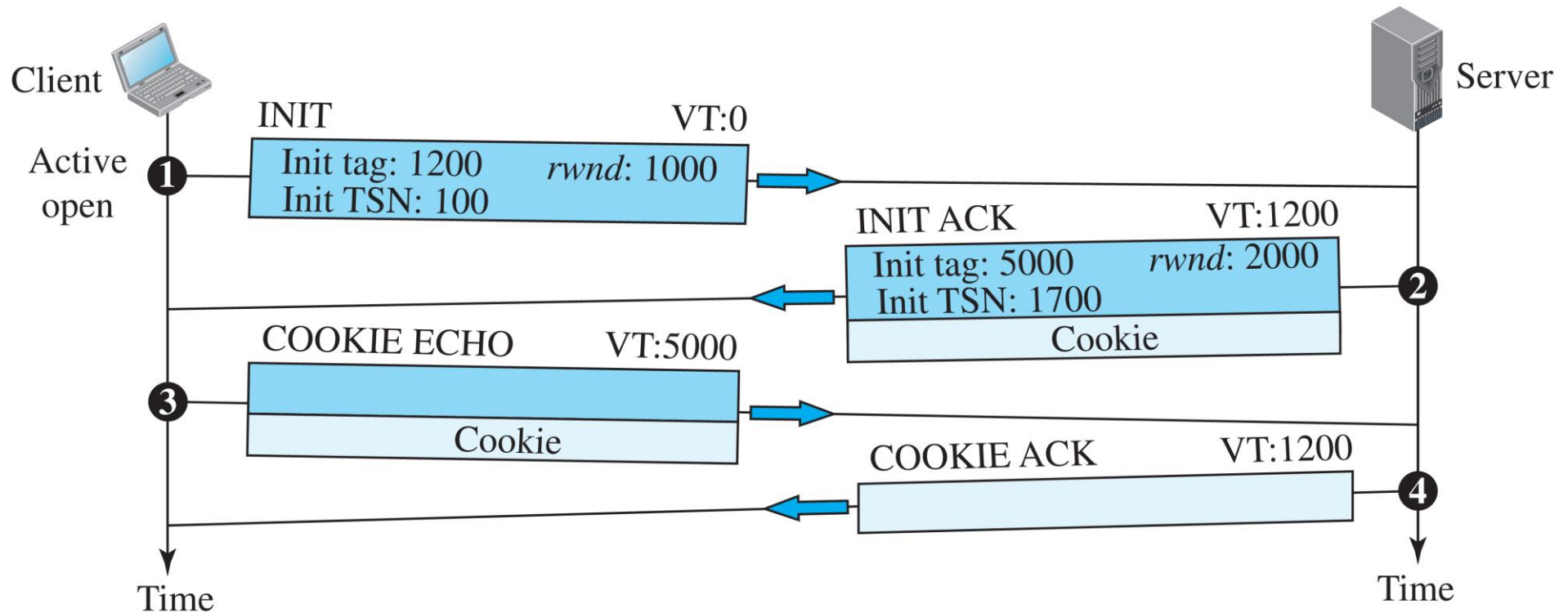
- Una conexión en SCTP se denomina una asociación.
- Se requieren tres fases para establecer una asociación
- Una asociación es simplemente una conexión que soporta multihomming.
- Las tres fases son:
 - Establecimiento de asociación
 - Transferencia de datos
 - Terminación de asociación.



Una Asociación SCTP: Establecimiento de asociación

1. El servidor anuncia que éste esta listo para conectar (*passive open*)
2. El cliente emite una solicitud con un *active open*
3. SCTP comienza un *proceso de negociación de cuatro pasos (four-way handshaking)*

Proceso de negociación de 4 pasos (four-way handshaking)



Una Asociación SCTP: Establecimiento de asociación

- No se permite ningun *chunk* en un paquete que transporta un chunk INIT o INIT ACK.
- Un chunk COOKIE ECHO o COOKIE ACK puede transportar *data chunks*.

Una Asociación SCTP: Cookie

- Para prevenir ataques SYN flooding: SCTP genera un cookie.
- El cookie se envía con el segundo paquete a la dirección recibida en el primer paquete.
- Hay dos situaciones potenciales:
 1. Si el emisor del primer paquete es un atacante:
 - El servidor nunca recibe el tercer paquete
 - El cookie se pierde y no se asignan recursos

Una Asociación SCTP: Cookie

2. Si el emisor del primer paquete es un cliente honesto que necesita establecer una conexión:
 - El receptor recibe el segundo paquete, con el cookie.
 - Este envía un paquete (tercero en la serie) con el cookie, sin cambios.
 - El servidor recibe el tercer paquete y sabe que este proviene de un cliente honesto porque el cookie que el emisor envió está ahí.
 - El servidor ahora puede asignar recursos.
- Esta estrategia trabaja si ninguna entidad puede “comerse” la cookie “horneada” por el servidor.
 - Para garantizar esto, el servidor crea un compendio
 - La información y el compendio juntos hacen el cookie

Una Asociación SCTP: *chunks* de iniciación

- *Chunk* de iniciación de asociación.

Association Initiation Chunk Field Use		
Initiate Tag		This 32-bit tag is exchanged during association initialization and is placed on every message that applies to the session. It is used to help prevent security breaches and to validate that individual packets apply to this instance of the association. The tag must not have value zero.
Advertised Receiver Window Credit		The initial size of the receiver window—that is, the number of bytes that the sender may send. This value may be modified by Selective Acknowledgement chunks.
Number of Outbound Streams		Defines the maximum number of outbound streams the sender of this chunk wants to create in this association. A value of zero must not be used.
Number of Inbound Streams		Defines the maximum number of inbound streams the sender of this chunk is willing to allow the receiver to create in this association. A value of zero must not be used.
Initial Transmission Sequence Number (TSN)		The initial TSN is the sequence number that identifies the first byte of data that will be sent on the association. Any number in the range 0 to 4,294,967,295 is acceptable. Some implementations randomize this value and set it to the value of the Initiate Tag field.
Parameter Type	Parameter Name	Use
5	IPv4 Address	One of the IPv4 addresses that may be used to identify the sender's end of the association. Multiple IPv4 and IPv6 addresses may be present. If no addresses are present, the SCTP application should use the address from the IP datagram that delivered the SCTP packet.
6	IPv6 Address	One of the IPv6 addresses that may be used to identify the sender's end of the association. Multiple IPv4 and IPv6 addresses may be present. If no addresses are present, the SCTP application should use the address from the IP datagram that delivered the SCTP packet.
9	Cookie Preservative	A value in milliseconds by which the sender is suggesting that the cookie timeout value be increased to prevent the cookie expiring again (as it has just done) during association establishment.
11	Host Name Address	A single host name that may be used to identify the sender's end of the association. The host name may not be present along with any IPv4 or IPv6 addresses, and only one host name may be used.
12	Supported Address Types	The address types that the sender supports and from which the receiver may choose addresses for its end of the association. If this parameter is absent, the sender supports all address types.

- Campos en un *chunk* de iniciación de asociación

- Parámetros opcionales en un *chunk* de iniciación de asociación

Chunk Type = 1 (Initiation)	Flags (Reserved)	Chunk Length = 66
Initiate Tag		
Advertised Receive Window Credit		
Number of Outbound Streams		Number of Inbound Streams
Initial Transmission Sequence Number		
Optional Parameter Type = 4 (IPv4 Address)		Parameter Length = 8
IPv4 Address		
Optional Parameter Type = 5 (IPv6 Address)		Parameter Length = 20
IPv6 Address		
IPv6 Address (continued)		
IPv6 Address (continued)		
IPv6 Address (continued)		
Optional Parameter Type = 9 (Cookie Preservative)		Parameter Length = 8
Suggested Cookie Life Span Increment (milliseconds)		
Optional Parameter Type = 12 (Support Address Types)		Parameter Length = 10
Address Type 4 (IPv4)		Address Type 5 (IPv6)
Address Type 11 (Host Name)		Padding

Una Asociación SCTP: *chunks* de iniciación

- *Chunk* de confirmación de iniciación de asociación.

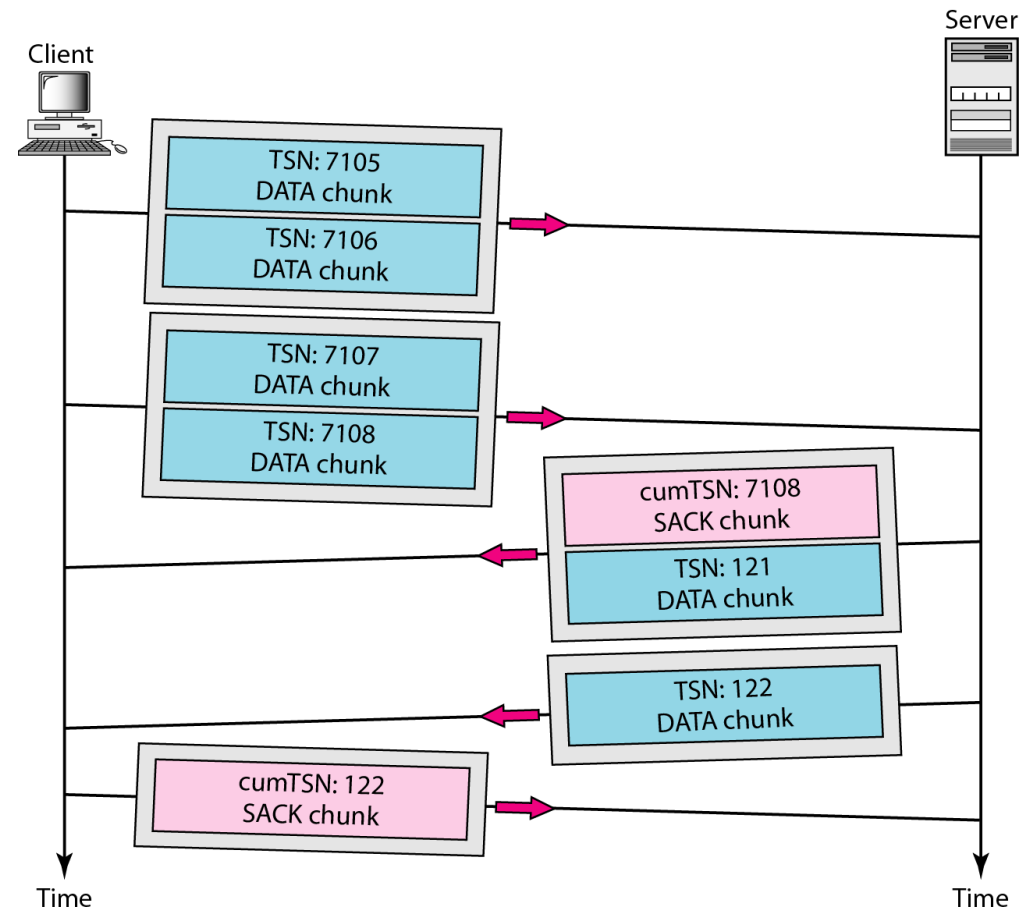
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Chunk Type = 2 (Initiation Ack)								Flags (Reserved)								Chunk Length															
Initiate Tag																															
Advertised Receive Window Credit																															
Number of Outbound Streams																Number of Inbound Streams															
Initial Transmission Sequence Number																															
Mandatory Parameter Type = 7 (State Cookie)																Parameter Length															
State Cookie																															
Optional Parameter Type = 11 (Host Name Address)																Parameter Length															
Host Name																															

Transferencia de datos

- Un mensaje recibido de un proceso se convierte en un *DATA chunk* o *chunks* si se fragmenta.
- En SCTP, solo los *DATA chunk* consumen TSNs;
- Los *DATA chunk* son los únicos que pueden ser confirmados.
- La confirmación en SCTP define el TSN del último *data chunk* recibido en orden.
- La confirmación en SCTP es acumulativa

Chunk Type = 0 (Data)	Reserved	U	B	E	Chunk Length
Transmission Sequence Number (TSN)					
Stream Identifier				Stream Sequence Number	
Payload Protocol Identifier					
User Data					

Transferencia de datos: Trasterferencia de datos simple



Transferencia de datos: Transferencia de datos Multihoming

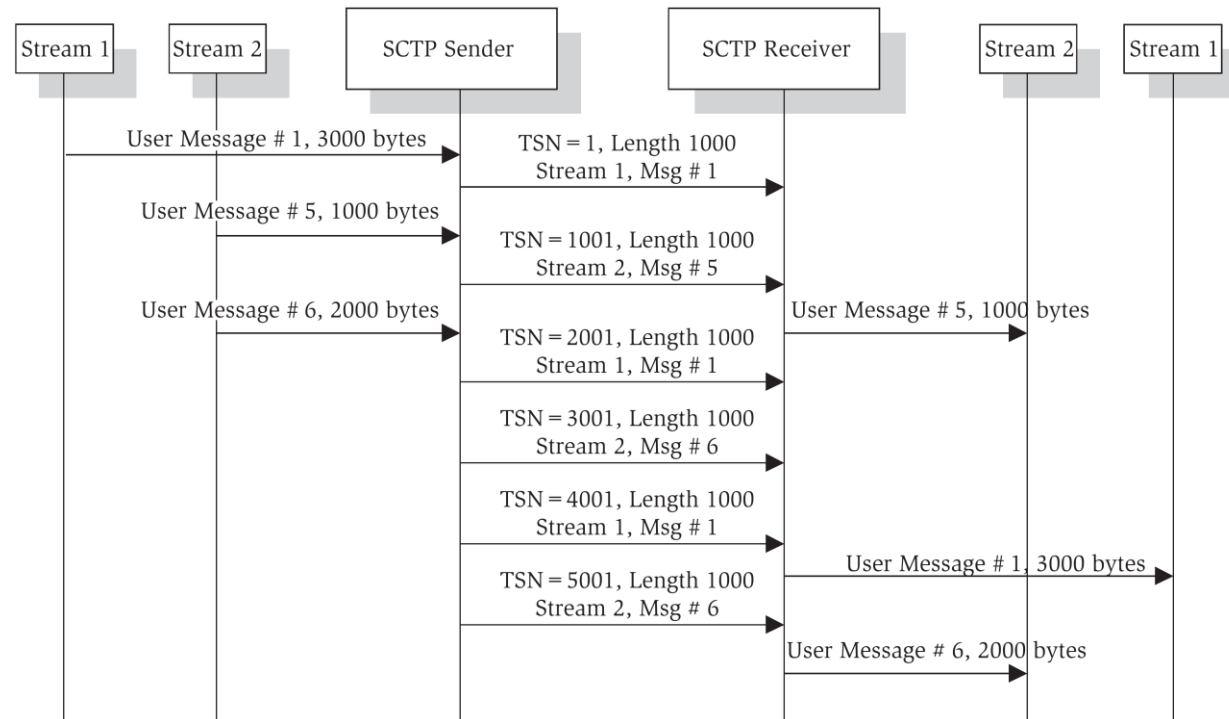
- Multihomming permite a ambos extremos definir múltiples direcciones IP para la comunicación.
- Solo una de esas direcciones puede definirse como la dirección primaria, el resto son direcciones alternativas.
- La dirección primaria se define durante el establecimiento de la asociación.
- Una fuente define la dirección primaria para un destino.

Transferencia de datos: Entrega multiflujo

- SCTP usa números TSN para manejar la transferencia de datos, el movimiento de *data chunks* entre la fuente y el destino.
- La entrega de los *data chunks* es controlada por los SI y los SSN.
- SCTP puede soportar múltiples flujos y un mensaje puede pertenecer a uno de esos flujos.
- A cada flujo se le asigna un identificador de flujo (SI) que define unívocamente a ese flujo

Transferencia de datos: Entrega multiflujo

- Multiplexado y demultiplexado de flujos de datos de usuario en una asociación única SCTP



Transferencia de datos: Fragmentación

- El tamaño de un datagrama IP transportando un mensaje puede ser determinado mediante la suma del tamaño del mensaje, en bytes, a los cuatro encabezados:
 - Encabezado de *data chunk*.
 - Chunks SACK necesarios.
 - Encabezado SCTP general.
 - Encabezado IP.
- Si el tamaño total excede el MTU, el mensaje necesita ser fragmentado.

Transferencia de datos: Heartbeat

- Al abrirse una asociación, SCTP puede probar periódicamente si esta permanece establecida y activa.
- Para este fin usa los *chunk* Heartbeat y Heartbeat Acknowledgement

0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Chunk Type = 4 (Heartbeat)								Flags (Reserved)								Chunk Length															
Heartbeat Info Type = 1 (Sender Info)																Heartbeat Info Length															
Sender Heartbeat Info																															

Chunk Type = 5 (Heartbeat Ack)								Flags (Reserved)								Chunk Length															
Heartbeat Info Type = 1 (Sender Info)																Heartbeat Info Length															
Sender Heartbeat Info																															

Transferencia de datos: Aborto y error de operación

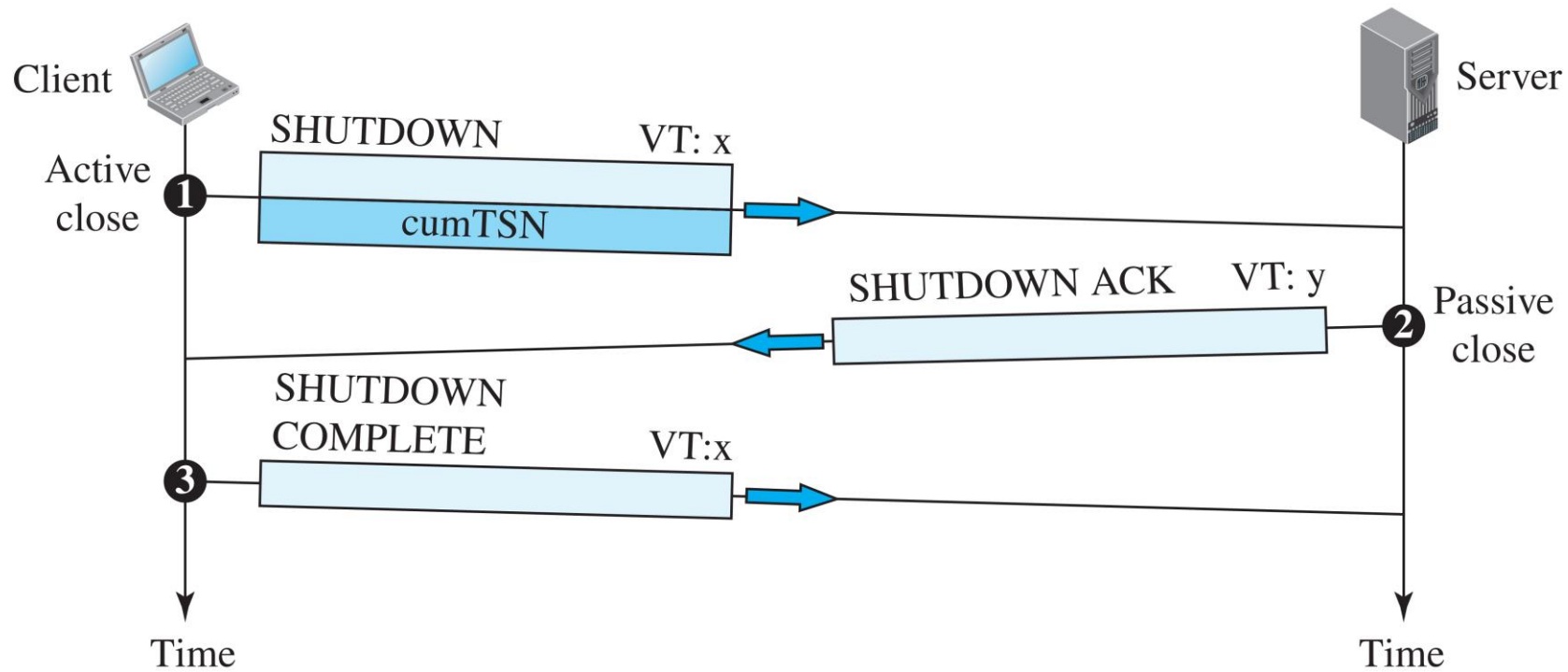
- Un *chunk* de Aborto permite terminar una asociación si se detecta algún error.
- Un *chunk* de Error de Operación permite notificar errores no fatales

Chunk Type = 6 (Abort)	Reserved	T	Chunk Length
First Cause Code		First Cause Length	
First Cause Data			
Other Cause Parameters			
Last Cause Code		Last Cause Length	
Last Cause Data			

Chunk Type = 9 (Operation Error)	Flags (Reserved)	Chunk Length
First Cause Code		First Cause Length
First Cause Data		
Other Cause Parameters		
Last Cause Code		Last Cause Length
Last Cause Data		

Cause	Meaning	Additional Information
1	Invalid Stream Identifier	The value of the invalid stream identifier was received in a data chunk.
2	Missing Mandatory Parameter	This is a count of missing mandatory parameters and the parameter type number of each missing parameter.
3	Stale Cookie Error	A cookie has been received in a State Cookie Echo chunk but the cookie has expired by the number of microseconds indicated. Note that this value is in microseconds even though the Suggested Cookie Life Span Increment given by the Cookie Preservative chunk is in milliseconds.
4	Out of Resource	No data is passed when this error is reported.
5	Unresolvable Address	The complete unresolvable address is passed encoded as an SCTP parameter so that its type and length can be seen.
6	Unrecognized Chunk Type	This error returns the chunk type, flags, and length of the unrecognized chunk.
7	Invalid Mandatory Parameter	This error is returned when one of the mandatory parameters on an Association Initiate or Initiate Acknowledgement chunk is set to an invalid value. No data is returned with this error, so it is not possible for the sender to determine which parameter is at fault.
8	Unrecognized Parameters	This error returns the full SCTP parameter that is unrecognized.
9	No User Data	A data chunk (see below) was received with a valid TSN but no data was present. This error returns the TSN that was received.
10	Cookie Received While Shutting Down	No data is passed when this error is reported.

Terminación de Asociación



SCTP no permite una situación de cierre a medias

Terminación de Asociación: chunks

- *Chunks* de terminación de asociación

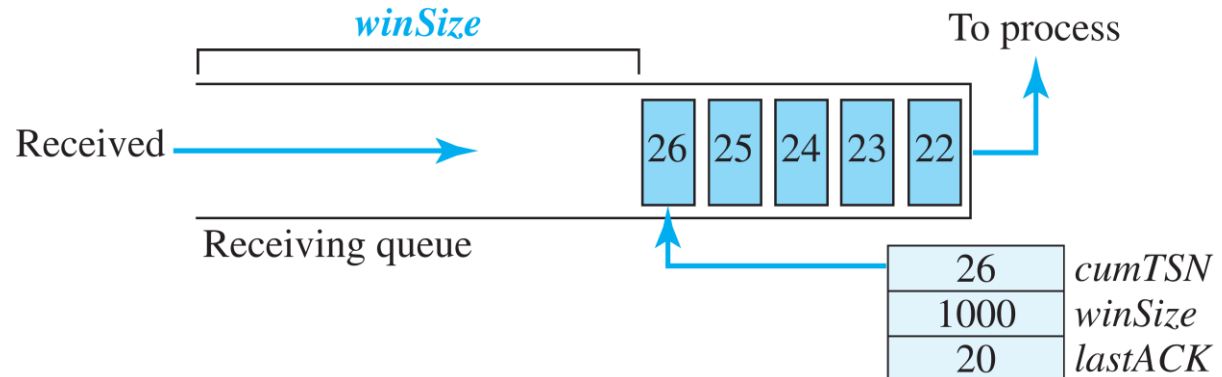
Chunk Type = 7 (Shutdown)	Flags (Reserved)	Chunk Length = 8
Cumulative Transmission Sequence Number Acknowledgement		

Chunk Type = 8 (Shutdown Ack)	Flags (Reserved)	Chunk Length = 4
----------------------------------	---------------------	------------------

Chunk Type = 14 Shutdown Complete	Reserved	T	Chunk Length = 4
--------------------------------------	----------	---	------------------

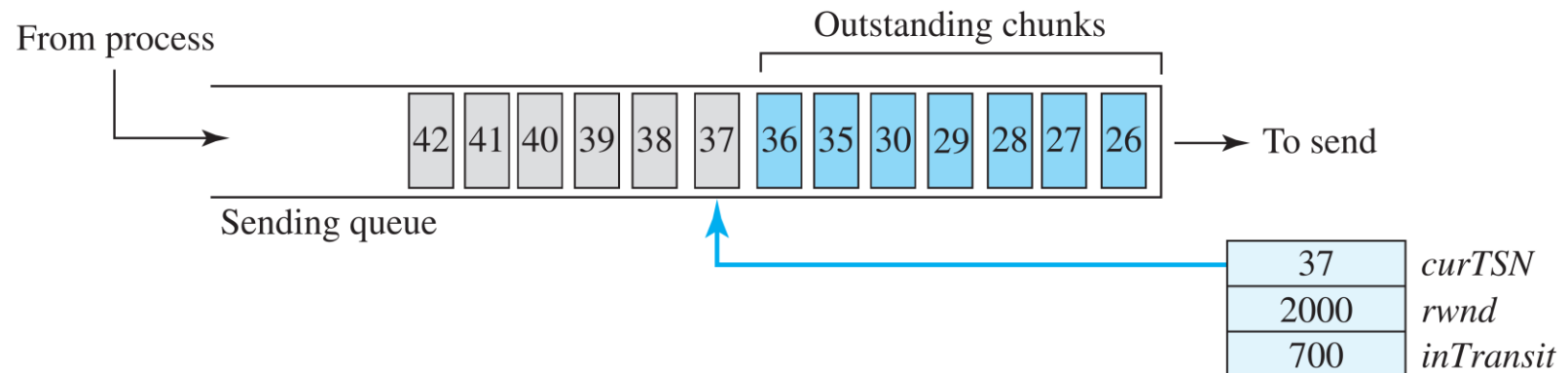
Control de flujo: Lado del receptor

- Cuando el sitio recibe un data chunk, lo almacena al final del buffer y resta el tamaño del chunk de winSize. El TSN del chunk se almacena en cumTSN.
- Cuando el proceso lee un chunk, lo retira de la cola y suma el tamaño del chunk retirado a winSize,
- Cuando el receptor decide enviar un SACK, este verifica el valor de lastACK; si este es menor que cumTSN, este envia un SACK con un numero TSN acumulativo igual a cumTSN. Tambien incluye el valor de winSize como el tamaño de ventana anunciado. El valor de lastACK se actualiza al valor de cumTSN

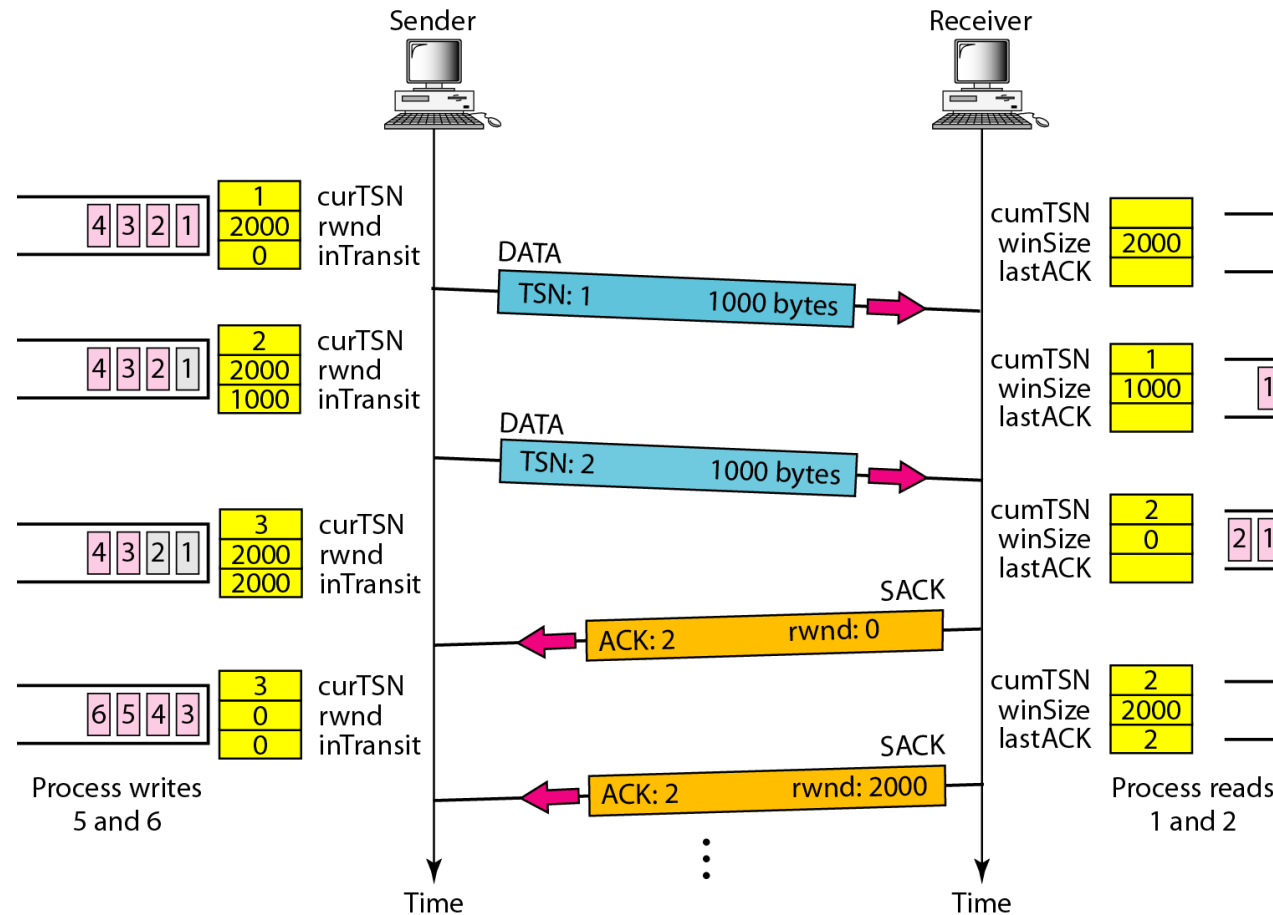


Control de flujo: Lado del emisor

- Un chunk aputado por $curTSN$ puede ser enviado si el tamaño de los datos es menor o igual a la cantidad ($rwnd - inTransit$). Después de enviar el chunk, el valor de $curTSN$ se incrementa en 1 y ahora apunta al siguiente al siguiente chunk que se enviará. El valor de $inTransit$ se incrementa en el tamaño de los datos en el chunk transmitido.
- Cuando se recibe un SACK, los chunk con un TSN menor o igual al TSN acumulativo en el SACK se retiran de la cola y se descartan. El emisor no tiene que preocuparse de estos mas. El valor de $inTransit$ se reduce en el tamaño total de los chunk descartados. El valor de $rwnd$ se actualiza con el valor de la ventana anunciado en el SACK.

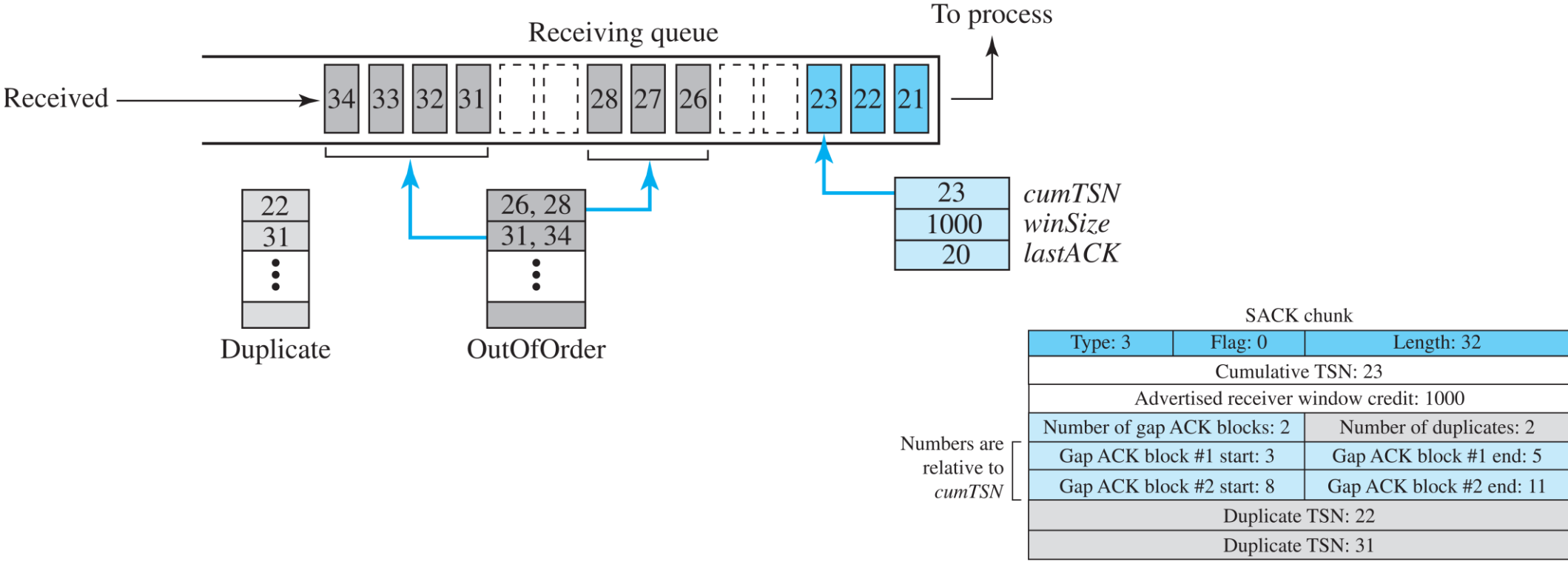


Escenario de control de flujo

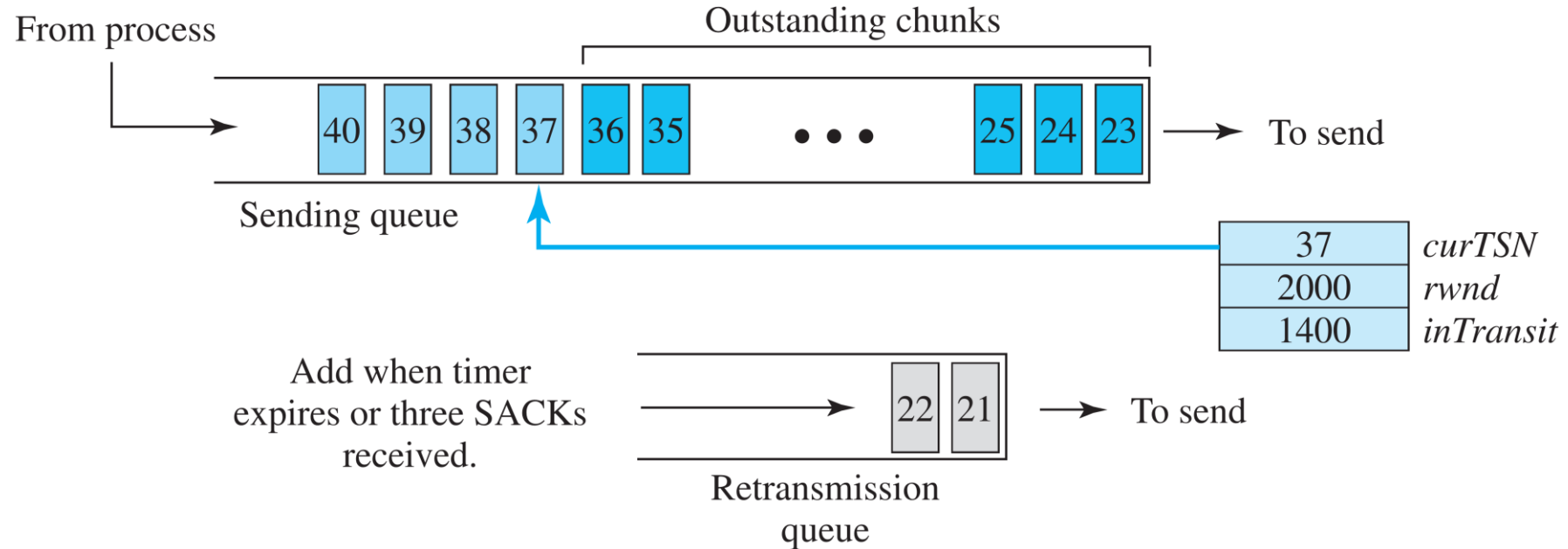


Control de errores: Lado del receptor

- Utiliza un chunk SACK para reportar el estado del buffer del receptor al emisor

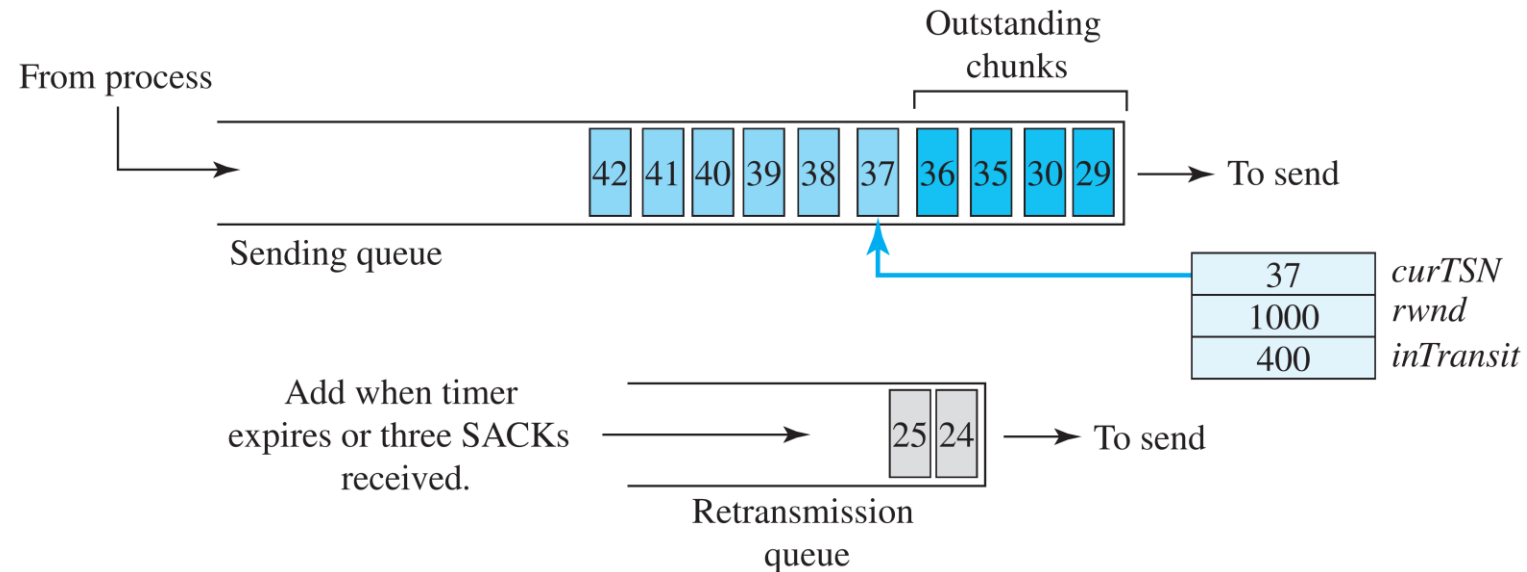


Control de errores: Lado del emisor



Control de errores: Lado del emisor

- Nuevo estado del emisor al recibir un chunk SACK

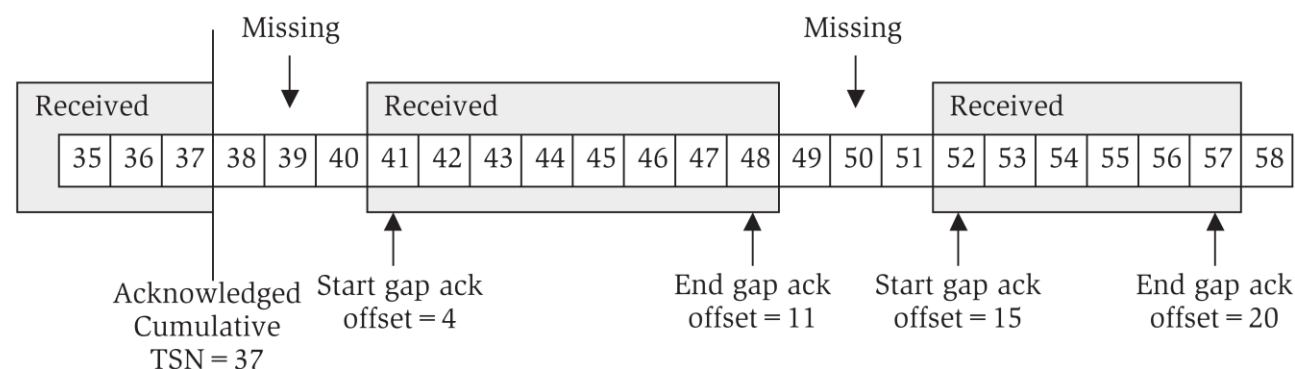


Control de errores: Retransmisión

- Para controlar un chunk perdido o descartado, SCTP utiliza dos estrategias:
 - Usando timers de retransmisión (como en TCP)
 - Recibiendo cuatro SACKs por el mismo *chunk* perdido.

Control de errores: SACK

- SACK indica que todos los bytes hasta el TSN incluido se recibieron correctamente.
- SACK informa de los datos recibidos, así como de los bloques faltantes



Chunk Type = 3 (Selective Ack)	Flags (Reserved)	Chunk Length
Cumulative Transmission Sequence Number Acknowledgement		
Advertised Receiver Window Credit		
Number of Gap Blocks		Number of Duplicate TSNs
First Gap Block Start		First Gap Block End
Last Gap Block Start		Last Gap Block End
First Duplicate TSN		
Last Duplicate TSN		

Control de congestionamiento

- Se utiliza las mismas estrategias que en TCP

Referencias

- Farrel A. “*The Internet and its Protocols. A Comparative Approach*”.
- Forouzan B. “*Data Communications and Networking with TCP/IP Protocol Suite*”. 6 edition 2022.