

实验6.3 TCP拥塞控制-1

滑动窗口与零窗口 探查机制分析

TCP的流量控制和拥塞控制

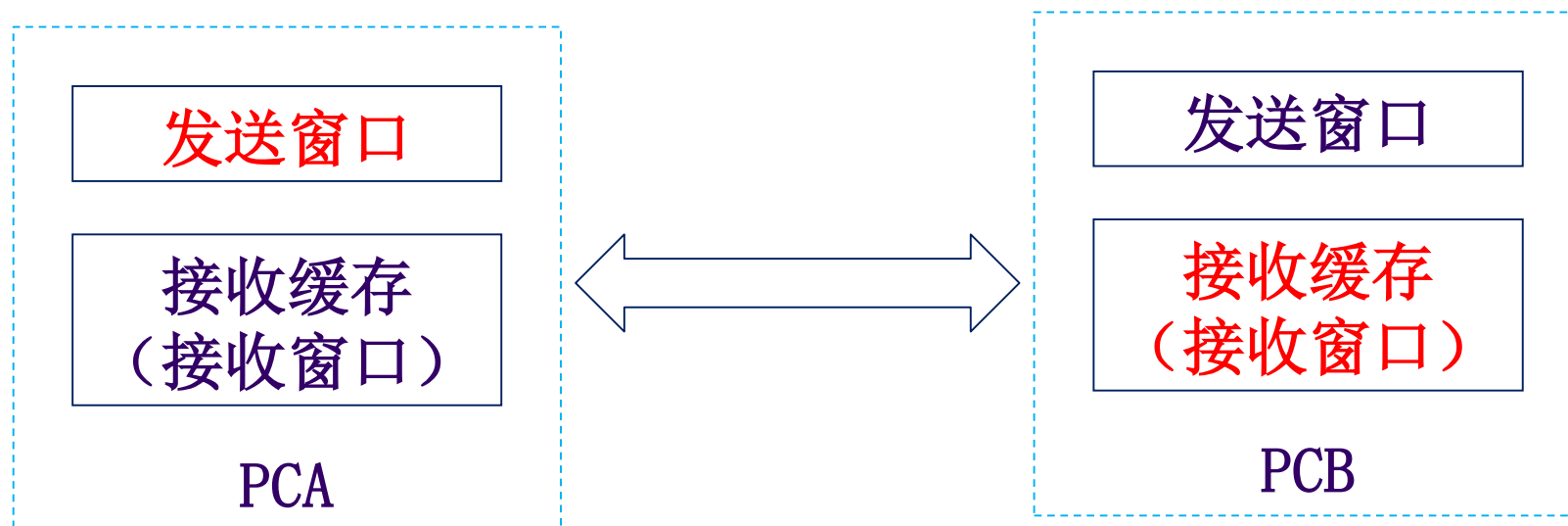
- **TCP “字节流” 的概念（逻辑）**
- **流：编了序号的“字节串”**
- **可靠性：面向连接、数据传输ACK机制**
- **流量控制：使用滑动窗口机制**
- **窗口仅仅是在流上的一个连续片断**
- **拥塞控制：是通过TCP中端到端基于滑窗的流量控制机制完成的**
- **拥塞控制四算法：慢启动、拥塞避免、快重传、快恢复**

流量控制和拥塞控制的区别和关系

- **流量控制**主要考虑**接收端**，目的是使发送端的发送速率不超过接收端的接收能力；
- **拥塞控制**主要考虑端节点之间的**网络环境**，目的是使负载不超过网络的传送能力；
- 在很长一段时间内，接收端驱动的**TCP**流量控制机制是唯一可行的拥塞控制方法；
- 但“流量控制”和“拥塞控制”两者切不可混淆，实际上，前者只是实现后者的一个技术实现途径而已。

TCP数据传输

- 双向同时发送和接收数据
- 捎带确认
- 本次实验，只做**PCA—>PCB**单向数据传输。

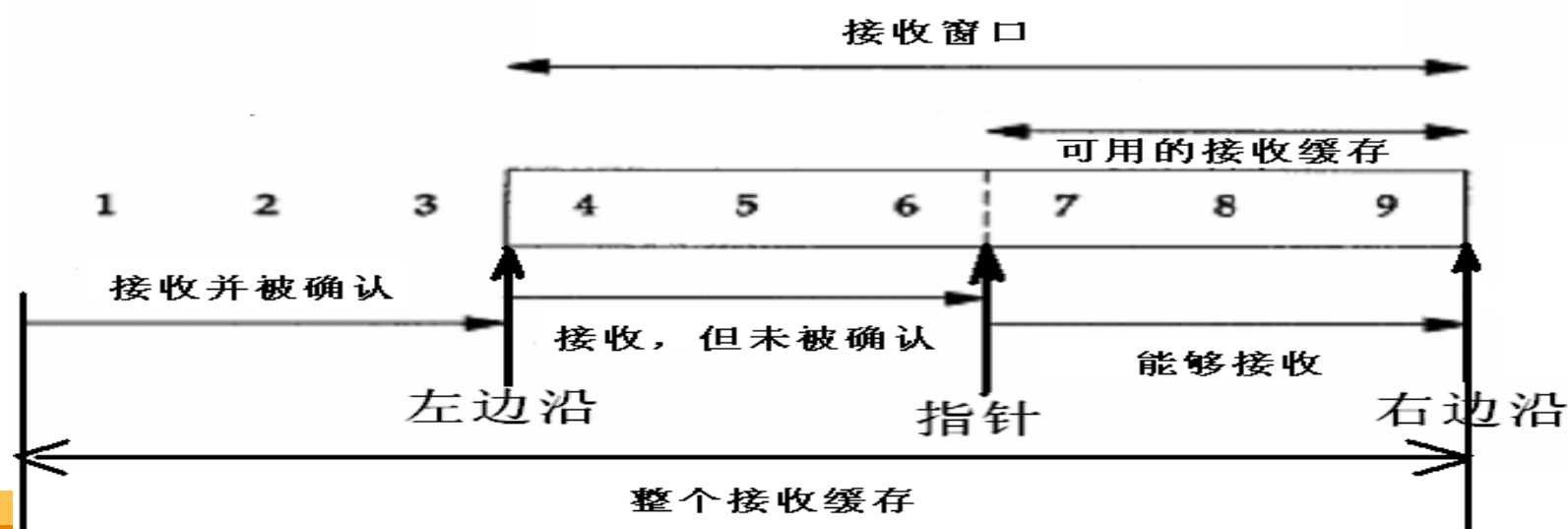


拥塞控制—滑动窗口

发送窗口



接收窗口



TCP相关参数说明

Show TCP sender's some values about sndwnd:

pkt_seqno	pkt_type	RorS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left
1	snd_con_syn	1	2147483647	2	0	1769176647
2	rcv_con_syn_ack	1	2147483647	2	65535	1769176648
3	snd_con_ack	2	2147483647	2	65535	1769176648
4	snd_data	3	2147483647	2	65535	1769176648
5	snd_data	4	2147483647	2	65535	1769176648
6	rcv_ack	2	2147483647	3	65535	1769179496

- (1) **pkt_type**: 报文的类型，共有**snd_con_syn**, **rcv_con_syn_ack**, **snd_con_ack**, **snd_data**, **rcv_ack**, **timeout**, **tx_full**等几种类型。
- (2) **RorS_seqno**: 发送报文序号或接收报文序号（按发送方发送和接收区分）。
- (3) **snd_ssthresh**: 发送方的慢启动算法向拥塞避免算法切换的门限值。
- (4) **snd_cwnd**: 发送方的拥塞窗口大小（以**MSS**为单位）。
- (5) **rcv_wnd**: 接收方通告的接收窗口大小（以字节为单位）。
- (6) **snd_wnd_left**: 发送窗口左边沿，即该序号之前的所有数据均已发送并收到确认。

TCP相关参数说明

<code>snd_wnd_pointer</code>	<code>snd_wnd_left+cwnd</code>	<code>snd_wnd_left+rcv_wnd</code>	<code>snd_wnd_pointer-left</code>
1769176648	1769179567	1769176647	1
1769176648	1769179568	1769242183	0
1769176648	1769179568	1769242183	0
1769178048	1769179568	1769242183	1400
1769179496	1769179568	1769242183	2848
1769179496	1769183876	1769245031	0

■ (7) `snd_wnd_pointer`: 发送窗口指针, 即已发送数据的最高序号+1。

■ (8) `snd_wnd_left+rcv_wnd`: 此值=`snd_wnd_left + rcv_wnd`, 即发送窗口的左边沿加上接收方通告的接收窗口大小 (字节) 后的值。

■ (9) `snd_wnd_left+cwnd`: 此值=`snd_wnd_left + snd_cwnd*1460`, 即发送窗口的左边沿加上发送方的拥塞窗口大小 (字节) 后的值。

■ (10) `snd_wnd_pointer-left`: 此值=`snd_wnd_pointer - snd_wnd_left`, 表明发送方已发送但未被确认的数据字节数。

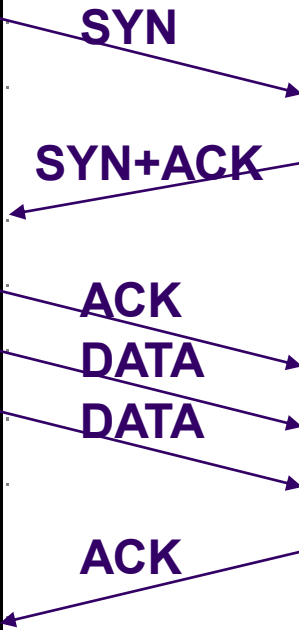
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400				
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520493721	2848				
6	rcv_ack	2	2147483647	3	65535	520431034	520431034	520435414	520496569	0				

客户

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106
520428186	520429586	520431106
520428186	520431034	520431106
等待确认.....		
520431034	520431034	520435414

服务器

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		
520428186	520428186	520493721
520428186	520429586	520493721
520428186	520431034	520493721
520431034	520431034	520496569



-----1-----2-----3-----4-----5-----6-----7-----8-----9-----0-----1-----2-----3-----4-----														
Show TCP sender's some valuces about sndwnd:														
pktsegno	pkt_type	Ror\$	segno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd			

客户（CLOSED）

发送方窗口		
左边沿	指针	右边沿

服务器（CLOSED）

接收方窗口		
左边沿	指针	右边沿

[illegible]

客户 (CLOSED)

[illegible]

服务器 (LISTEN)

[illegible]

接收窗口windows size=65535

-----1-----2-----3-----4-----5-----6-----7-----8-----9-----0-----1-----2-----3-----4-----										
Show TCP sender's some valuces about sndwnd:										
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1

客户 (SYN-SENT)		
发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		

服务器 (LISTEN)		
接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

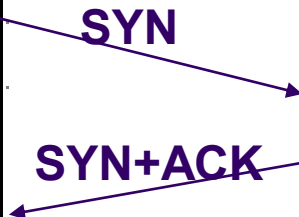
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				

客户 (SYN-SENT)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		

服务器 (SYN-RCVD)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

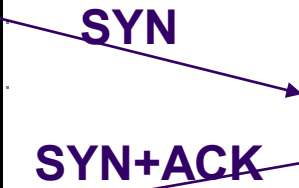
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				

客户 (SYN-SENT)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106

服务器 (SYN-RCVD)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

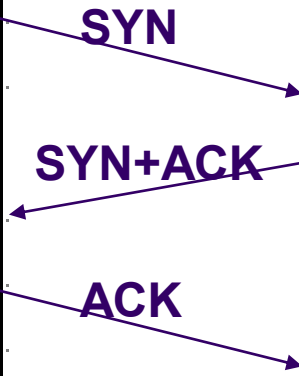
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				

客户 (ESTABLISHED)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106

服务器 (SYN-RCVD)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				

客户 (ESTABLISHED)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106

拥塞窗口cwnd=2*1460

服务器 (ESTABLISHED)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		
520428186	520428186	520493721

接收窗口windows size=65535

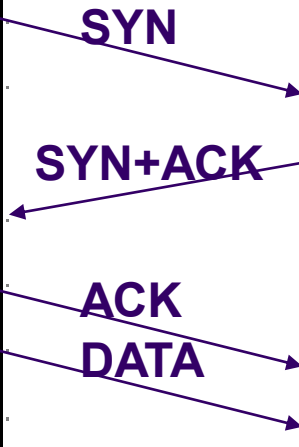
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400				

客户 (ESTABLISHED)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106
520428186	520429586	520431106

服务器 (ESTABLISHED)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		
520428186	520428186	520493721
520428186	520429586	520493721



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

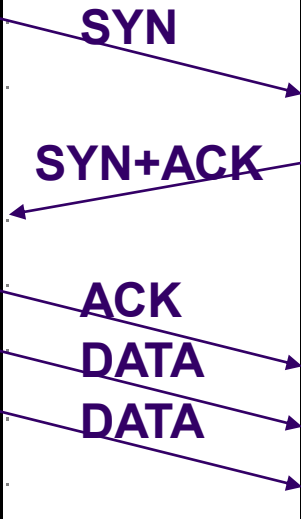
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	RcvS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400				
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520493721	2846				

客户 (ESTABLISHED)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106
520428186	520429586	520431106
520428186	520431034	520431106
等待确认.....		

服务器 (ESTABLISHED)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		
520428186	520428186	520493721
520428186	520429586	520493721
520428186	520431034	520493721



拥塞窗口cwnd=2*1460

接收窗口windows size=65535

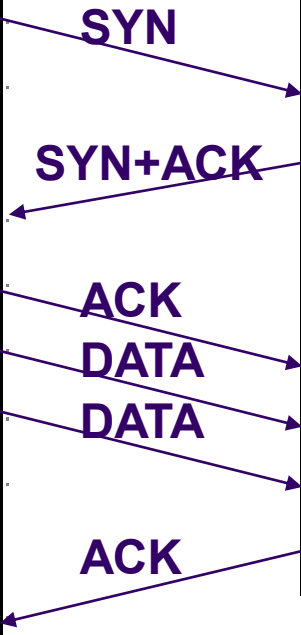
	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400				
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520493721	2848				
6	rcv_ack	2	2147483647	3	65535	520431034	520431034	520435414	520496569	0				

客户 (ESTABLISHED)

发送方窗口		
左边沿	指针	右边沿
主动发起连接 SYN.....		
520428185	520428186	
SYN 等待确认.....		
520428186	520428186	520431106
520428186	520428186	520431106
520428186	520429586	520431106
520428186	520431034	520431106
等待确认.....		
520431034	520431034	520435414

服务器 (ESTABLISHED)

接收方窗口		
左边沿	指针	右边沿
被动监听状态.....		
520428185	520428186	520493720
520428186	520428186	520493721
SYN+ACK		
SYN 等待确认.....		
520428186	520428186	520493721
520428186	520429586	520493721
520428186	520431034	520493721
520431034	520431034	520496569

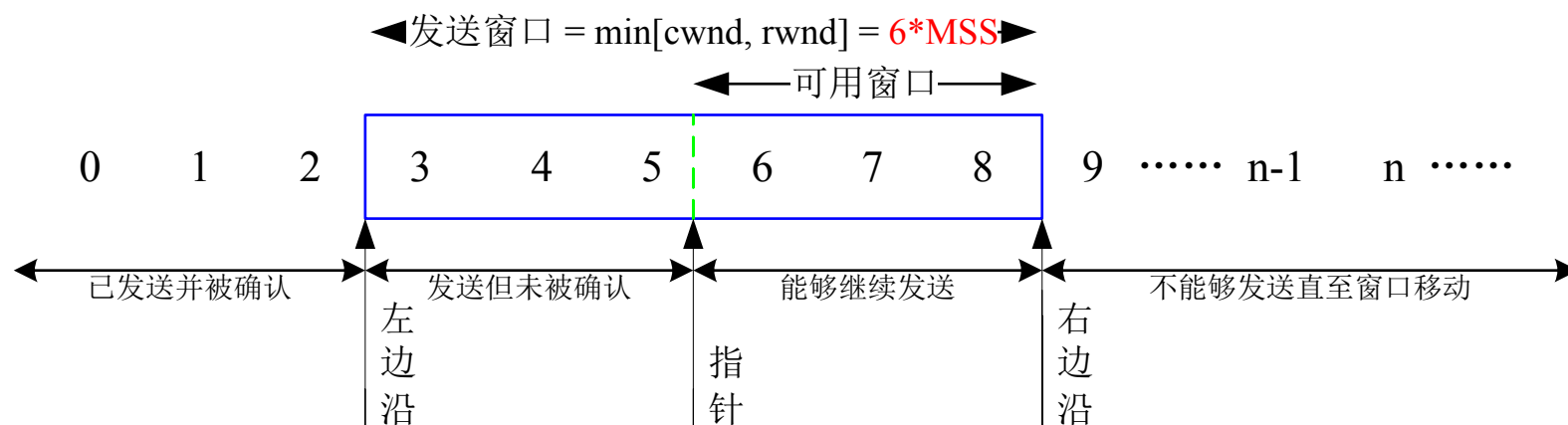


拥塞窗口cwnd=3*1460

接收窗口windows size=65535

发送窗口示意图—发送窗口大小

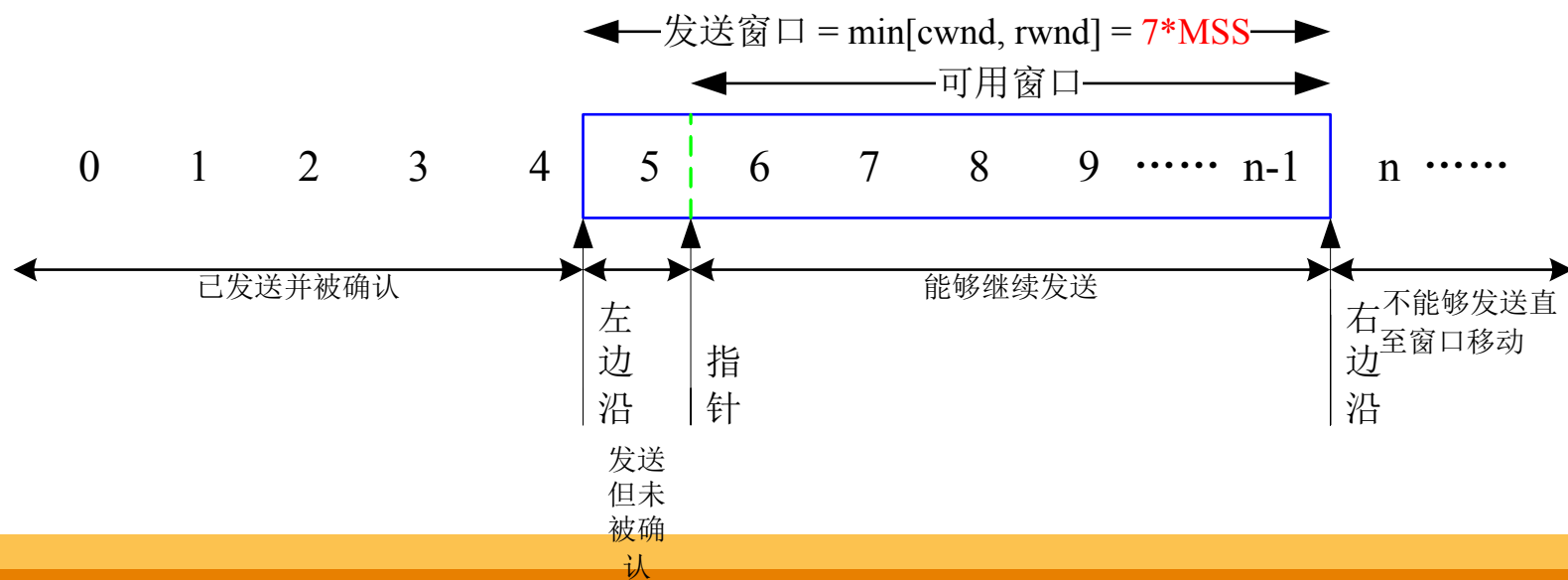
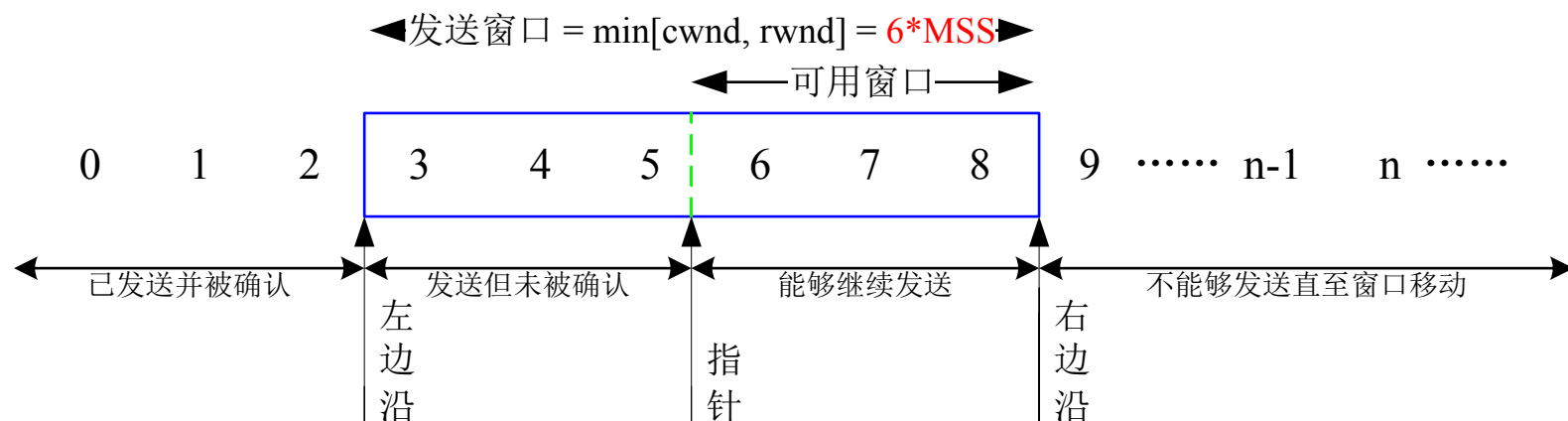
■ 发送窗口 $\leq \min[\text{snd_cwnd}, \text{rcv_wnd}]$



Show TCP sender's some values about sndwnd:

pkt_seqno	pkt_type	RorS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left
1	snd_con_syn	1	2147483647	2	0	1769176647
2	rcv_con_syn_ack	1	2147483647	2	65535	1769176648
3	snd_con_ack	2	2147483647	2	65535	1769176648
4	snd_data	3	2147483647	2	65535	1769176648
5	snd_data	4	2147483647	2	65535	1769176648
6	rcv_ack	2	2147483647	3	65535	1769179496

发送窗口示意图



发送窗口示意图—左边沿

左边沿=最近的一个Ack报文的Ack_SEQ_num

注：若当前报文是一个Ack报文，

则snd_wnd_left=Ack_SEQ_num

否则， snd_wnd_left=上一个Ack报文的Ack_SEQ_num

No. ,	Time	Source	Destination	Protocol	Info
46	700.656901	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=2849 Win=62687 Len=0
47	700.656945	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
48	700.656953	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
49	700.656961	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
50	700.658092	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=5745 Win=59791 Len=0
51	700.658123	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]

Transmission Control Protocol, Src Port: 1234 (1234), Dst Port: 33006 (33006), Seq: 1, Ack: 5745, Len: 0						
Source port: 1234 (1234)						
Destination port: 33006 (33006)						
Sequence number: 1 (relative sequence number)						
Show TCP sender's some valuces about sndwnd:						
pkt_seqno	pkt_type	RorS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left
1	snd_con_syn	1	2147483647	2	0	1769176647
2	rcv_con_syn_ack	1	2147483647	2	65535	1769176648
3	snd_con_ack	2	2147483647	2	65535	1769176648
4	snd_data	3	2147483647	2	65535	1769176648
5	snd_data	4	2147483647	2	65535	1769176648
6	rcv_ack	2	2147483647	3	65535	1769179496

发送窗口示意图—指针

指针=最近的一个发送数据报文的SEQ_num+Len(也就是Wireshark软件显示的Next_SEQ_num)

注：若当前报文是一个发送数据报文，
则 $snd_wnd_pointer = Next_SEQ_num = SEQ_num + Len$
否则， $snd_wnd_pointer =$ 上一个发送报文的Next_SEQ_num

No.	Time	Source	Destination	Protocol	Info
46	700.656901	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=2849 Win=62687 Len=0
47	700.656945	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
48	700.656953	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
49	700.656961	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
50	700.658092	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=5745 Win=59791 Len=0
51	700.658123	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]

Internet Protocol Version 4, Src: 192.168.2.2, Dst: 192.168.1.2

Transmission Control Protocol, Src Port: 1234 (1234), Dst Port: 33006 (33006), Seq: 1, Ack: 5745, Len: 0

Source port: 1234 (1234)

Destination port: 33006 (33006)

Sequence number: 1 (relative sequence number)

Show TCP window state and sequence numbers

snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd_pointer-left
1769176648	1769179567	1769176647	1
1769176648	1769179568	1769242183	0
1769176648	1769179568	1769242183	0
1769178048	1769179568	1769242183	1400
1769179496	1769179568	1769242183	2848
1769179496	1769183876	1769245031	0

发送窗口示意图—右边沿

右边沿 = 左边沿 + 发送窗口大小

= min[snd_wnd_left+cwnd, snd_wnd_left+rcv_wnd]

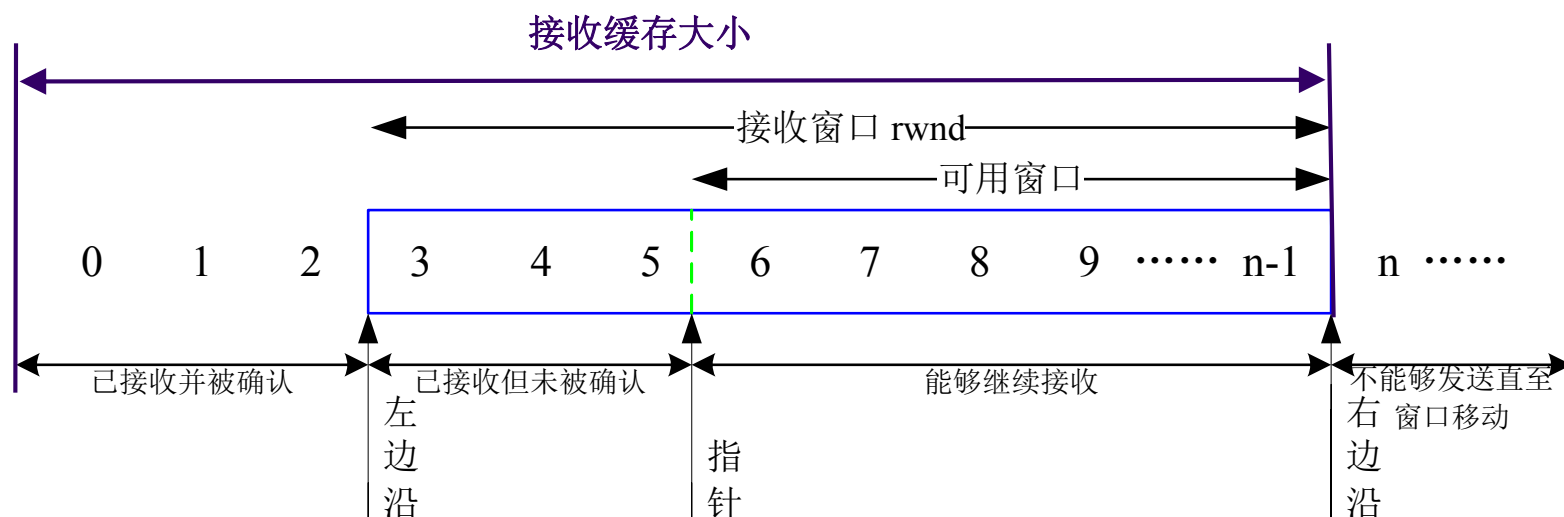
No. .	Time	Source	Destination	Protocol	Info
46	700.656901	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=2849 Win=62687 Len=0
47	700.656945	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
48	700.656953	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
49	700.656961	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
50	700.658092	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=5745 Win=59791 Len=0
51	700.658123	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]

Internet Protocol, Src: 192.168.2.2 (192.168.2.2), Dst: 192.168.1.2 (192.168.1.2)					
[-] Transmission Control Protocol, Src Port: 1234 (1234), Dst Port: 33006 (33006), Seq: 1, Ack: 5745, Len: 0					
Source port: 1234 (1234)					
Destination port: 33006 (33006)					
Sequence number: 1 (relative sequence number)					
Acknowledgement number: 5745 (relative ack number)					
Header length: 32 bytes					
[-] Flags: 0x0010 (ACK)					
Window size: 59791					
Checksum: 0x3407 [correct]					

snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd_pointer-left
1769176648	1769179567	1769176647	1
1769176648	1769179568	1769242183	0
1769176648	1769179568	1769242183	0
1769178048	1769179568	1769242183	1400
1769179496	1769179568	1769242183	2848
1769179496	1769183876	1769245031	0

接收窗口示意图—接收窗口大小与接收缓存

通告的接收窗口大小 \leq 接收缓存的大小



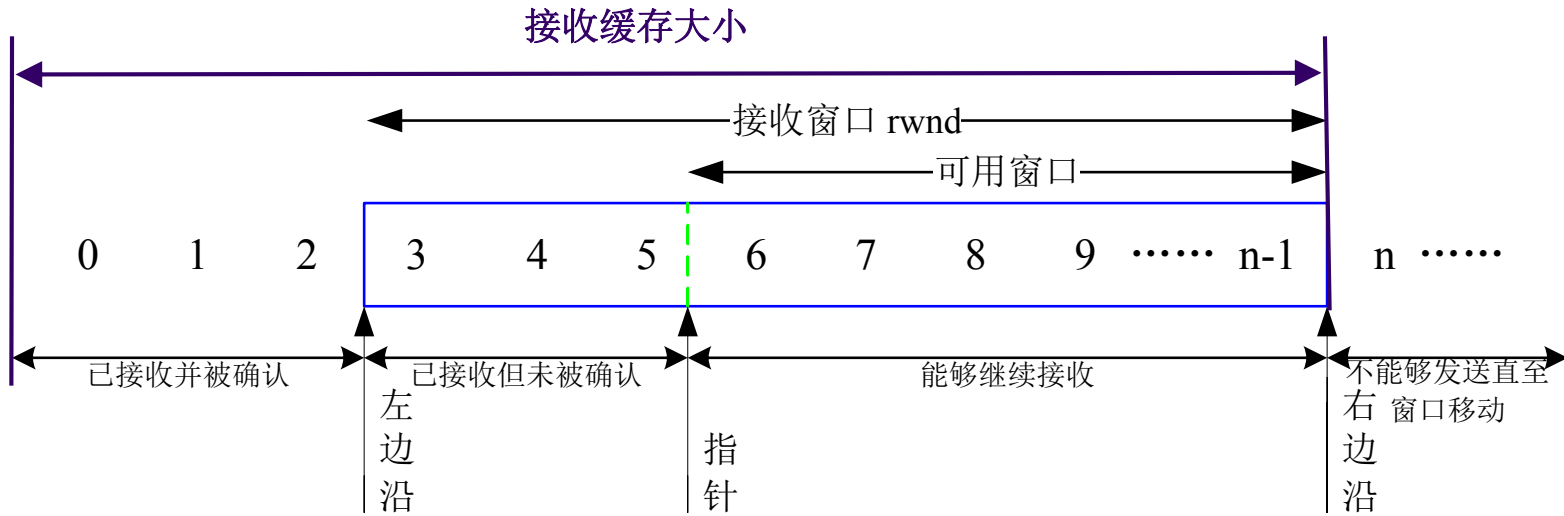
接收窗口被放入一个环形的接收缓存中，并在其中滑动；

Show TCP sender's some values about sndwnd:

pkt_seqno	pkt_type	RcvS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left
1	snd_con_syn	1	2147483647	2	0	1769176647
2	rcv_con_syn_ack	1	2147483647	2	65535	1769176648
3	snd_con_ack	2	2147483647	2	65535	1769176648
4	snd_data	3	2147483647	2	65535	1769176648
5	snd_data	4	2147483647	2	65535	1769176648

接收窗口示意图—左边沿

左边沿=最近的一个Ack报文的Ack_SEQ_num



No.	Time	Source	Destination	Protocol	Info
46	700.656901	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=2849 Win=62687 Len=0
47	700.656945	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
48	700.656953	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
49	700.656961	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
50	700.658092	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=5745 Win=59791 Len=0
51	700.658123	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]

Internet Protocol, Src Port: 1234 (1234), Dst Port: 33006 (33006), Seq: 1, Ack: 5745, Len: 0

Source port: 1234 (1234)

Destination port: 33006 (33006)

Sequence number: 1 (relative sequence number)

Acknowledgement number: 5745 (relative ack number)

Header length: 32 bytes

Flags: 0x0010 (ACK)

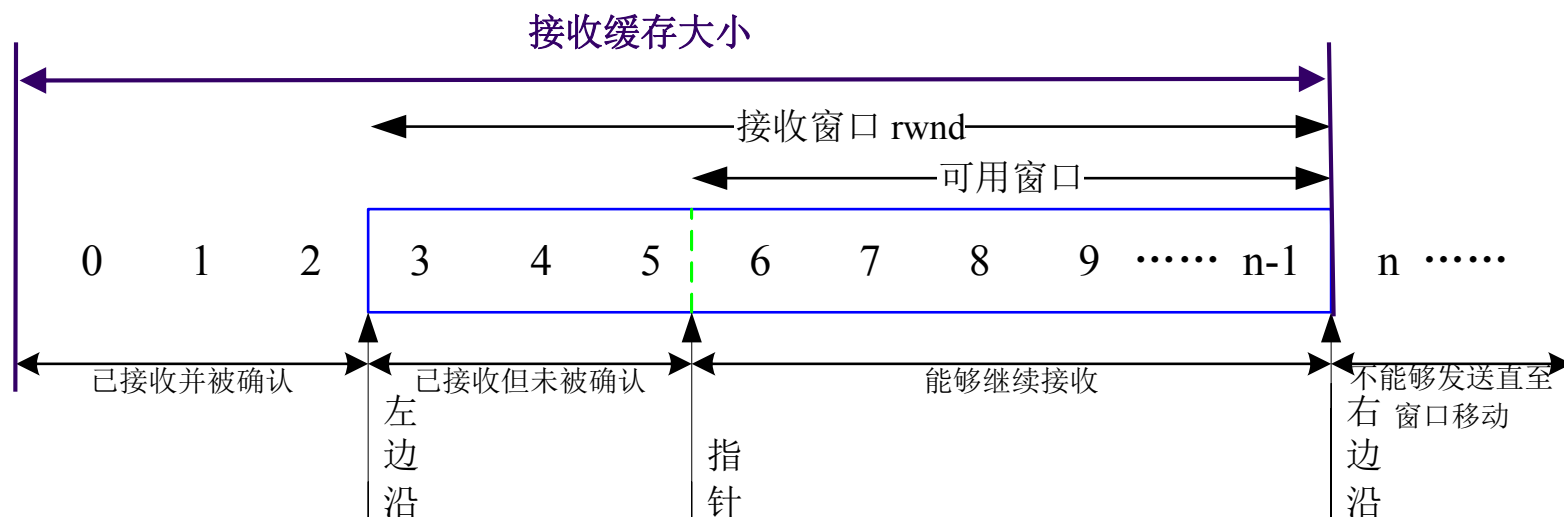
Window size: 59791

Checksum: 0x3407 [correct]

Options: (12 bytes)

接收窗口示意图—指针

■ 指针 = 最近的一个发送报文的SEQ_num+Len (也就是Wireshark中显示的Next_SEQ_num)

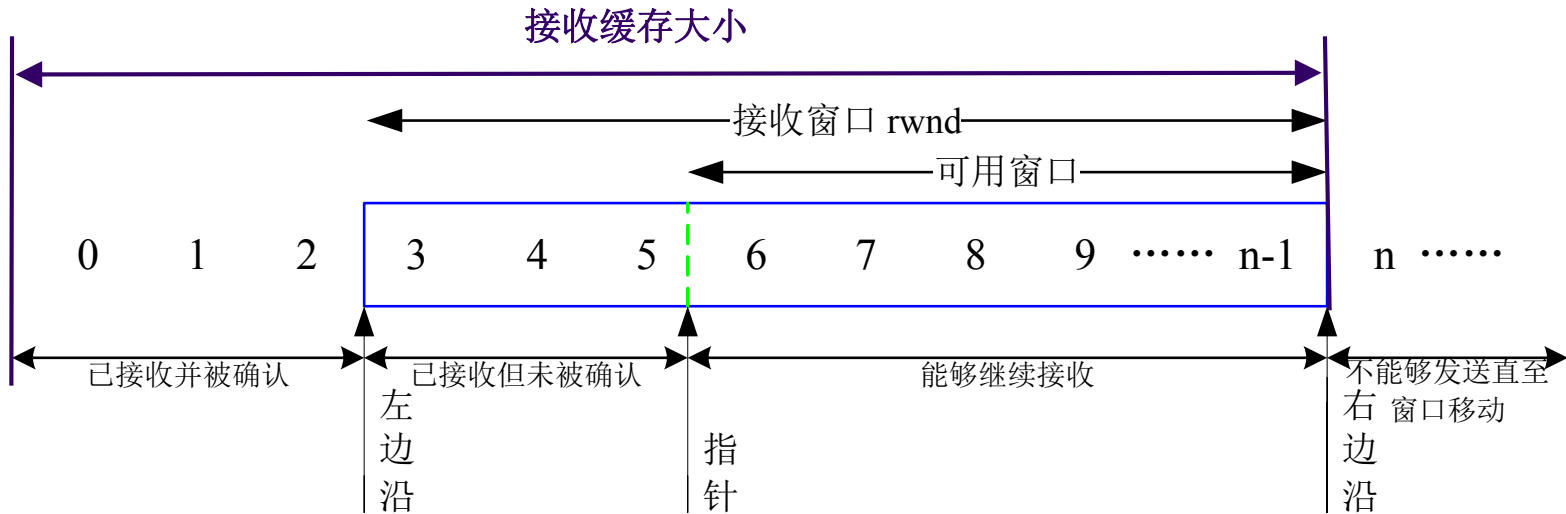


No. ↓	Time	Source	Destination	Protocol	Info
46	700.656901	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=2849 Win=62687 Len=0
47	700.656945	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
48	700.656953	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
49	700.656961	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]
50	700.658092	192.168.2.2	192.168.1.2	TCP	1234 > 33006 [ACK] Seq=1 Ack=5745 Win=59791 Len=0
51	700.658123	192.168.1.2	192.168.2.2	TCP	[TCP segment of a reassembled PDU]

Transmission Control Protocol, Src Port: 33006 (33006), Dst Port: 1234 (1234), Seq: 4297, Ack: 1, Len: 1448					
Source port: 33006 (33006)					
Destination port: 1234 (1234)					
Sequence number: 4297 (relative sequence number)					
[Next sequence number: 5745 (relative sequence number)]					
Acknowledgement number: 1 (relative ack number)					
Header length: 32 bytes					
Flags: 0x0010 (ACK)					
Window size: 5840					
Checksum: 0xe72a [correct]					

接收窗口示意图—右边沿

右边沿 = 左边沿 + 接收窗口大小



如果进程读取了部分数据，右边沿即向前滑动了相应的距离，窗口变大。

snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd_pointer-left
1769176648	1769179567	1769176647	1
1769176648	1769179568	1769242183	0
1769176648	1769179568	1769242183	0
1769178048	1769179568	1769242183	1400
1769179496	1769179568	1769242183	2848

发送窗口小结

- 发送窗口 $\leq \min[\text{snd_cwnd}, \text{rcv_wnd}]$
- 左边沿 = 最近的一个Ack报文的Ack_SEQ_num
 - 注：若当前报文是一个Ack报文，
则 $\text{snd_wnd_left} = \text{Ack_SEQ_num}$
否则， $\text{snd_wnd_left} =$ 上一个Ack报文的Ack_SEQ_num
- 指针 = 最近的一个发送数据报文的SEQ_num+Len(也就是ethereal显示的Next_SEQ_num)
 - 注：若当前报文是一个发送数据报文，
则 $\text{snd_wnd_pointer} = \text{Next_SEQ_num} = \text{SEQ_num} + \text{Len}$
否则， $\text{snd_wnd_pointer} =$ 上一个发送报文的Next_SEQ_num
- 右边沿 = 左边沿 + 发送窗口大小
 $= \min[\text{snd_wnd_left} + \text{cwnd}, \text{snd_wnd_left} + \text{rcv_wnd}]$

接收窗口小结

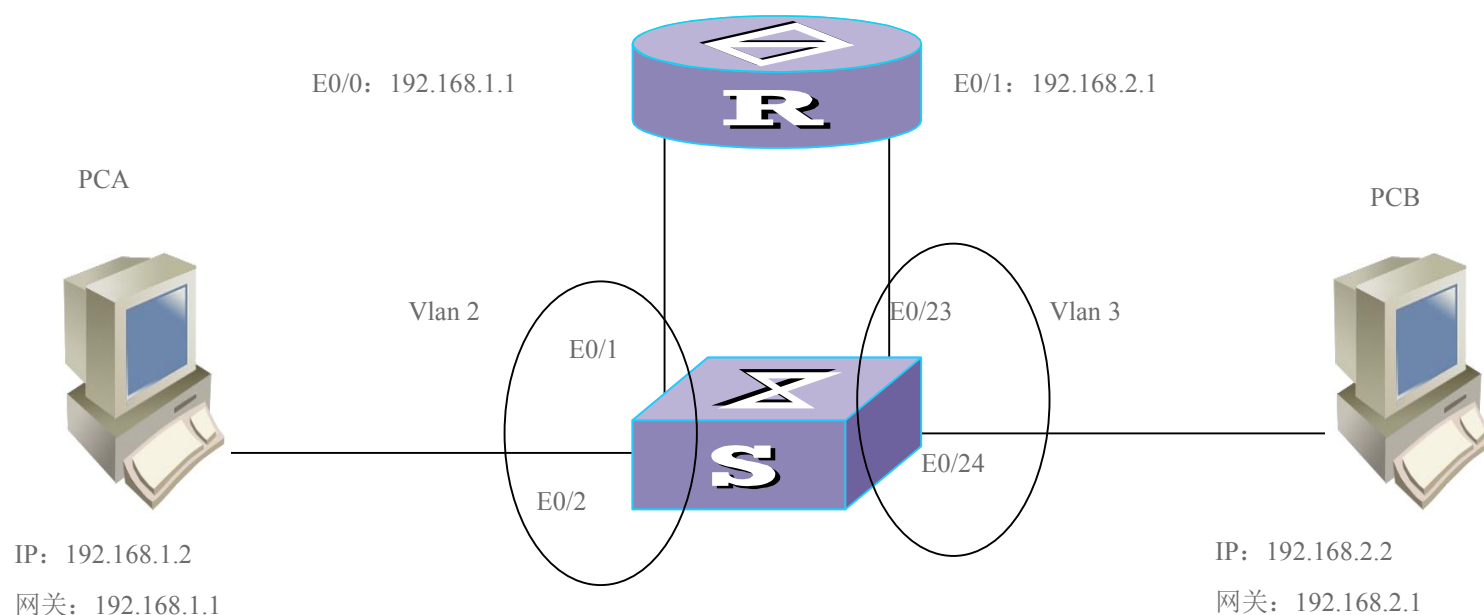
- 为了实验的分析方便，我们不妨设接收窗口如上图所示；
- 通告的接收窗口大小 \leq 接收缓存的大小；
- 接收窗口被放入一个环形的接收缓存中，并在其中滑动；
- 左边沿 = 最近的一个Ack报文的Ack_SEQ_num
- 指针 = 最近的一个发送报文的SEQ_num+Len(也就是Wireshark软件显示的Next_SEQ_num)
- 右边沿 = 左边沿 + 接收窗口大小
- 如果进程取走了部分数据，右边沿即向前滑动了相应的距离，窗口变大。

滑动窗口小结

- 窗口的左边沿是由接收方发送**ACK**报文驱动变化的；
 - 接收方发送**ACK**报文后其接收窗口左边沿向右滑动；
 - 发送方在收到**ACK**报文后其发送窗口左边沿也随之向右滑动。
- 窗口的指针是由发送方发送**DATA**报文驱动变化的；
 - 发送方发送**DATA**报文后其发送窗口指针向右滑动；
 - 接收方在收到**DATA**报文后其接收窗口指针也随之向右滑动。

组网图

- 网络连接图，所有PC进入Linux系统
- 2人一小组，用交换机和路由器各1台。



组网图一在线实验平台

Applications Places System ?

连线组网

实验组网 一键还原 帮助

PCA PCB PCC PCD R1 R2 S1 S2 Internet 主机联网

连接情况

端口1	端口2	flag
PCA-Eth0	S1-E0/1	
S1-E0/2	R1-E0/0	
PCB-Eth0	S1-E0/23	
S1-E0/24	R1-E0/1	

请选择抓包主机 : NULL 请选择被监听的端口 : NULL

监听端口 被监听端口

类型 : NULL s1 : NULL s2 : NULL

permit vlan id NULL tagged NULL untagged

类型 端口1 端口2 permit tagged untagged

选择聚合端口 : NULL NULL

s1端口1 s1端口2 s2端口1 s1端口2

滑动窗口机制实验步骤

- 按照组网图进行组网，确保组网正确和路由器、PC接口IP地址配置正确。
- 在Linux系统下都已预设了IP地址，实验时可直接选用。
- 为了方便大家实验，所有需要在“终端命令行”中输入的命令，都被编辑成多个脚本文件，放置在桌面上，只需用鼠标选中双击运行。
- PCA（PCC）（即发送端）在Linux下“终端命令行”中运行脚本来初始化“TCPConnection实时监控模块”：
- PCA（PCC）（即发送端）在Linux下打开一个新的“终端命令行”，使用命令启动TCPTest程序。
- 配置发送端和接收端的TCPTest程序的参数，然后传输数据。
- 在PCA（PCC）（即发送端）的“终端命令行”中运行脚本来读取“TCPConnection实时监控模块”已记录的此TCP连接期间的相关参数数据：
- 分析文件send2-组座号（或receive2-组座号）和send2-组座号-tcpsndwnddata.txt，体会滑动窗口机制。

软件介绍

Tcp 协议测试

TCP协议测试软件

要发送的文件:

目的IP地址:

目的端口:

发送缓存:

滞留时间:

每次写入套接字的字节数:

Nagle算法:

信息反馈:

Tcp 协议测试

TCP协议测试软件

要写入的文件:

指定端口:

休眠时间:

计数器阈值:

接收缓存:

滞留时间:

每次读出套接字的字节数:

Nagle算法:

信息反馈:

1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:													
pktseqno	pkt_type	Ror\$_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd_left+rcv_wnd	snd_wnd_left+rcv_wnd	snd_wnd_left+rcv_wnd	snd_wnd_left+rcv_wnd
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	520428185	520428185	520428185	1
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520431106	520431106	520431106	520431106	0
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520431106	520431106	520431106	520431106	0
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520431106	520431106	520431106	520431106	1400
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520431106	520431106	520431106	520431106	2848
6	rcv_ack	2	2147483647	3	65535	520431034	520431034	520435414	520435414	520435414	520435414	520435414	0

4号报文 (sender---data--->receiver) ↵

↵	rcv_wnd↵	snd_wnd_left↵	snd_wnd_pointer↵	snd_wnd_left+cwnd和 snd_wnd_left+rcv_wnd↵	snd_wnd_point- left↵
发送方发出报文↵	65535↵	520428186↵	520429586↵	520431106↵ 520493721↵	1400↵
发送窗口右边沿↵	$= \min [\text{snd_wnd_left+cwnd}, \text{snd_wnd_left+rcv_wnd}] = 520431106↵$				
↵	通告的接收窗口↵	接收窗口左边沿↵	接收窗口指针↵	接收窗口右边沿↵	在接收缓存中的数据量 (即未确认的数据) ↵
接收方收到DATA前↵	65535↵	520428186↵	520428186↵	520493721↵	0↵
接收方收到DATA后↵	65535↵	520428186↵	520429586↵	520493721↵	1400↵

左边沿：相同（不变）

指针：数据报文发送和接收前后会有变化，变化大小为数据长度

右边沿：左边沿+窗口大小（不变）

	1	2	3	4	5	6	7	8	9	0	1	2	3	4
Show TCP sender's some valuces about sndwnd:														
pktseqno	pkt_type	RorS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd				
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1				
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0				
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400				
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520493721	2848				
6	rcv_ack	2	2147483647	3	65535	520431034	520431034	520435414	520496569	0				

5 号报文 (sender---data--->receiver) ↵

↵	rcv_wnd↵	snd_wnd_left↵	snd_wnd_pointer↵	snd_wnd_left+cwnd和 snd_wnd_left+rcv_wnd↵	snd_wnd_point- left↵
发送方发出报文↵	65535↵	520428186↵	520431034↵	520431106↵ 520493721↵	2848↵
发送窗口 右边沿↵	$= \min [\text{snd_wnd_left+cwnd}, \text{snd_wnd_left+rcv_wnd}] = 520431106↵$				
↵	通告的接收窗口↵	接收窗口左边沿↵	接收窗口指针↵	接收窗口右边沿↵	在接收缓存中的数据量 (即未确认的数据) ↵
接收方接到DATA前↵	65535↵	520428186↵	520429586↵	520493721↵	1400↵
接收方接到DATA后↵	65535↵	520428186↵	520431034↵	520493721↵	2848↵

pktseqno	pkt_type	RorS_seqno	snd_ssthresh	snd_cwnd	rcv_wnd	snd_wnd_left	snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd
1	snd_con_syn	1	2147483647	2	0	520428185	520428186	520431105	520428185	1
2	rcv_con_syn_ack	1	2147483647	2	65535	520428186	520428186	520431106	520493721	0
3	snd_con_ack	2	2147483647	2	65535	520428186	520428186	520431106	520493721	0
4	snd_data	3	2147483647	2	65535	520428186	520429586	520431106	520493721	1400
5	snd_data	4	2147483647	2	65535	520428186	520431034	520431106	520493721	2848
6	rcv_ack	2	2147483647	3	65535	520431034	520431034	520435414	520496569	0

6 号报文 (receiver ---ack---> sender) ↵

↵	通告的接收窗口↵	接收窗口左边沿↵	接收窗口指针↵	接收窗口右边沿↵	在接收缓存中的数据量 (即未确认的数据) ↵
接收方发出 ACK↵	65535↵	520431034↵	520431034↵	520496569↵	0↵
↵	rcv_wnd↵	snd_wnd_left↵	snd_wnd_pointer↵	snd_wnd_left+cwnd 和 snd_wnd_left+rcv_wnd↵	snd_wnd_point- left↵
发送方接到 ACK 后↵	65535↵	520431034↵	520431034↵	520435414↵ 520496569↵	0↵
发送窗口右边沿↵	= min [snd_wnd_left+cwnd, snd_wnd_left+rcv_wnd] = 520435414↵				
发送方接到 ACK 前↵	65535↵	520428186↵	520431034↵	520431106↵ 520493721↵	2848↵
发送窗口右边沿↵	= min [snd_wnd_left+cwnd, snd_wnd_left+rcv_wnd] = 520431106↵				

左边沿：确认报文发送和接收前后会有变化，变化值为数据长度
 指针：相同（不变）
 右边沿：左边沿+窗口大小（有变化）

TCP拥塞控制机制

- 滑动窗口机制
- 零窗口探查机制
- 慢启动算法分析
- 拥塞处理（超时）分析
- 拥塞避免算法分析
- 超时重传机制分析
- 快重传和快恢复算法
- 糊涂窗口综合症和Nagle算法

软件介绍

Tcp 协议测试

TCP协议测试软件

要发送的文件:

目的IP地址:

目的端口:

发送缓存:

滞留时间:

每次写入套接字的字节数:

Nagle算法:

信息反馈:

Tcp 协议测试

TCP协议测试软件

要写入的文件:

指定端口:

休眠时间:

计数器阈值:

接收缓存:

滞留时间:

每次读出套接字的字节数:

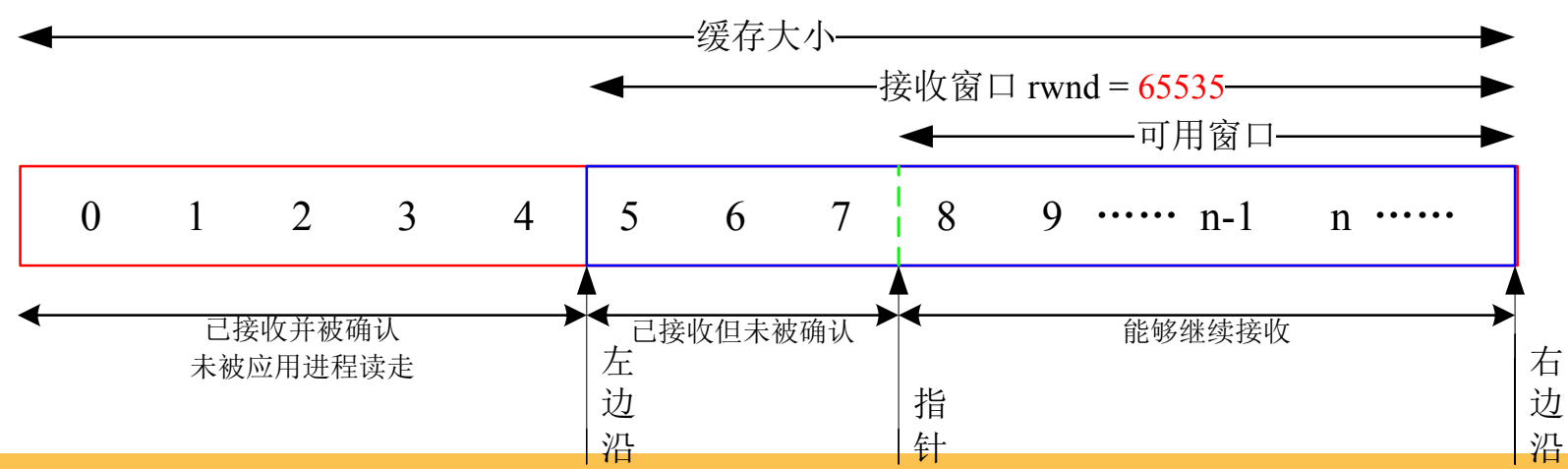
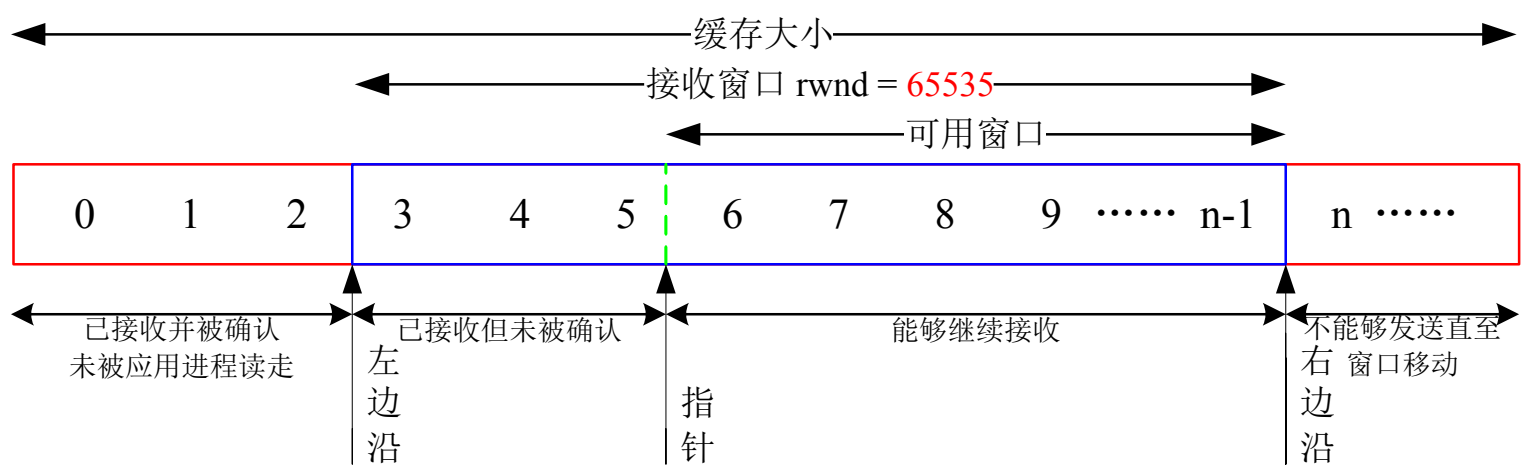
Nagle算法:

信息反馈:

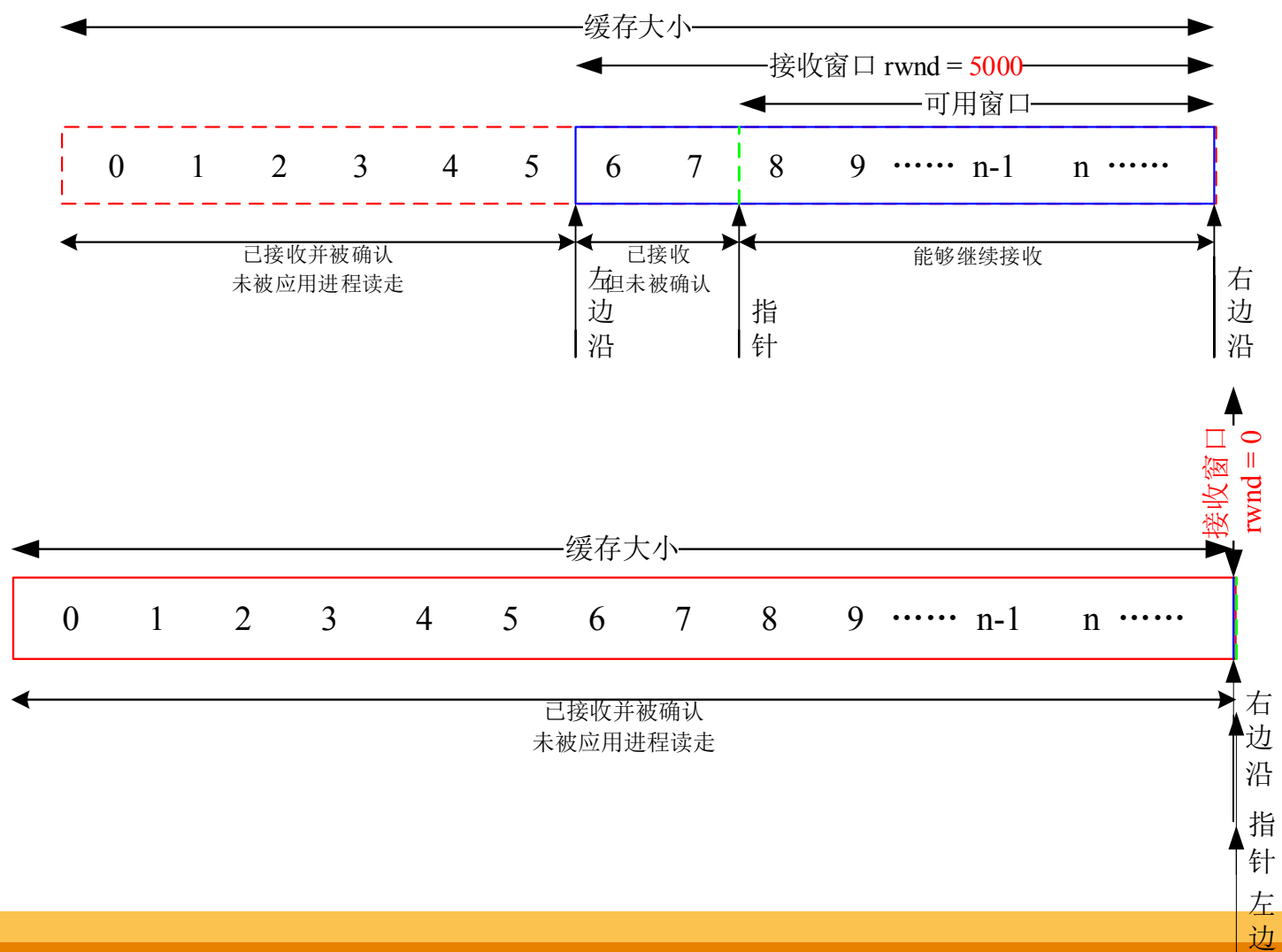
TCP协议测试软件

- 关于“**TCP**协议测试软件”的休眠设置的说明
 - 接收端中有“休眠时间”和“计数器阈值”两个与休眠有关的参数;
 - 休眠是指“**TCP**协议测试软件”应用程序进程在从套接字中读取 n 次（由“计数器阈值”设置）数据后开始休眠，即应用程序进程停止从套接字中读取数据;
 - 但此时处于内核态的“**TCP**协议模块”仍在工作

接收方窗口与缓存的关系

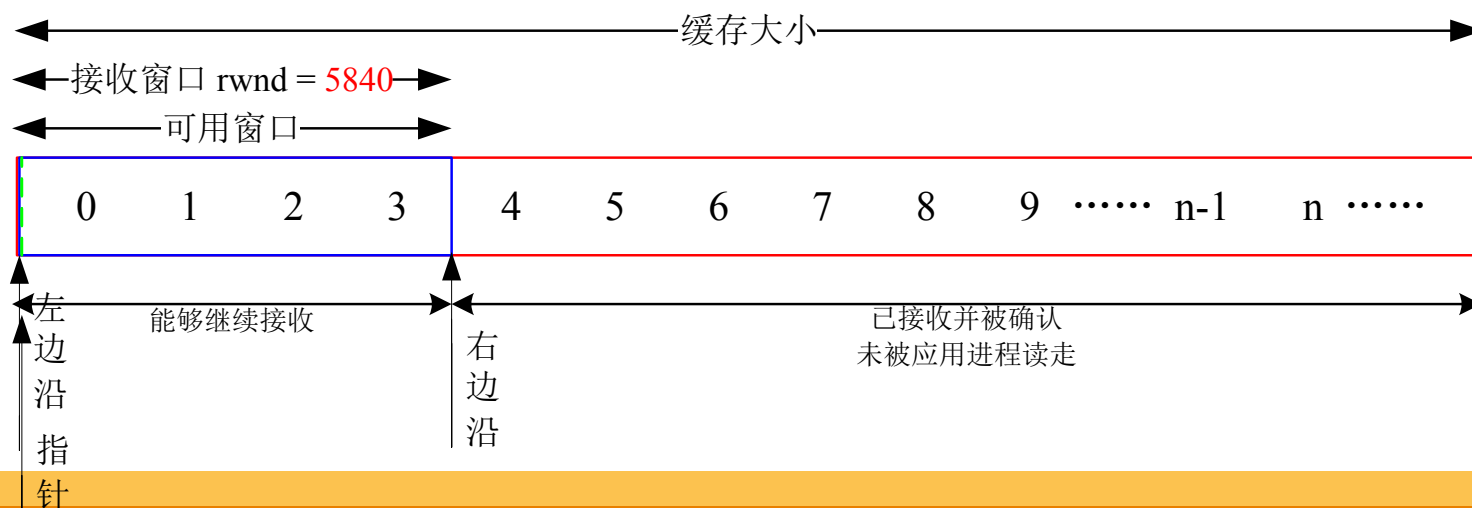


窗口合拢、窗口收缩、窗口张开（1）



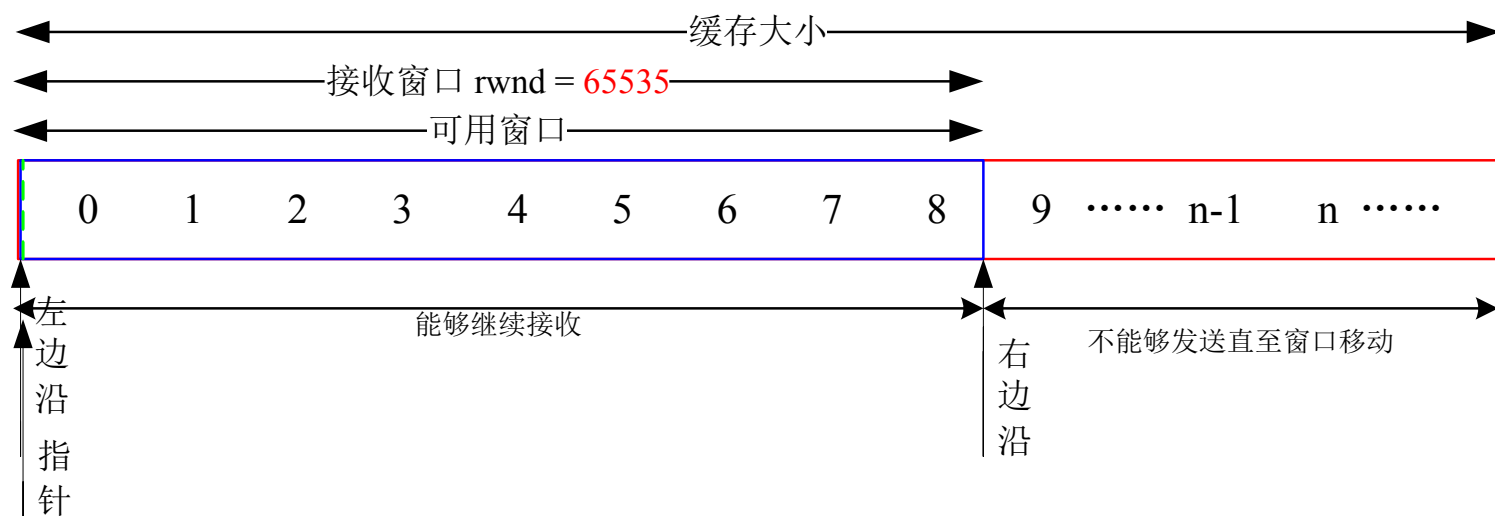
窗口合拢、窗口收缩、窗口张开（2）

- 窗口收缩定义：右边沿向左移动
- 窗口合拢定义：左边沿向右边沿靠近
- 窗口张开定义：右边沿向右移动



窗口合拢、窗口收缩、窗口张开（3）

- 窗口收缩一般不发生
- 窗口合拢发生在接收窗口持续减小期间
- 窗口张开发生在休眠结束后通告大窗口时



零窗口探查技术

- 当接收端接收缓存已满，将发送一个**0**窗口通告报文；
- 发送方收到后，等待非**0**窗口通告报文；
- 为防止因接收端的非**0**窗口通告报文丢失而产生死锁；
- 采用坚持定时器，定期发送窗口探查报文；维持**TCP**连接。
- 接收端收到探查报文后的立即响应

定时器时间的变化规律

- 窗口侦查报文指的是**Keep-Alive**报文
- 每相邻两条窗口侦查报文**Keep-Alive**报文时间差组成的数据序列的规律：成倍增加规律

窗口侦查报文发送或接收时间

m1	m2	m3	m4	m5	m6	m7
----	----	----	----	----	----	----	-------

相邻两条窗口侦查报文的时间差

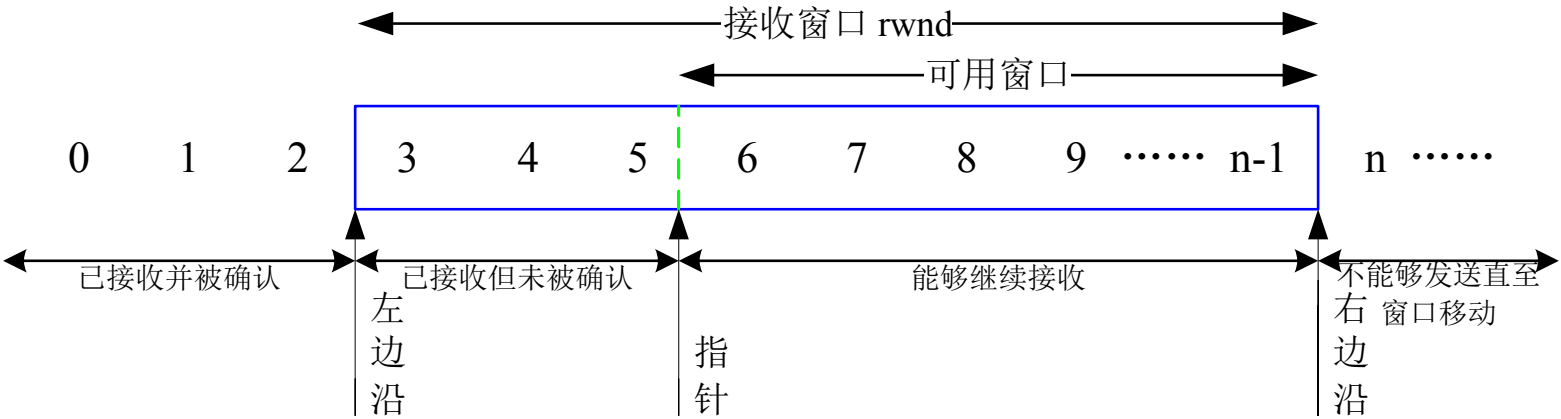
m2-m1 m3-m2 m4-m3 m5-m4 m6-m5 m7-m6

注意事项

- 结合报文写出对应接收端休眠、窗口收缩、窗口合拢、窗口张开的开始报文序号，并描述窗口的变化
 - 休眠开始：接收缓存，接收窗口开始单调减小直至0
 - 窗口收缩
- 零窗口探查机制
 - 时间规律
 - 探查与响应
 - 携带的数据与序号

接收窗口示意图

接收窗口是接收端缓冲区的一部分，接收窗口的大小由接收端缓冲区大小决定；
窗口由一个变量 ACK_seq_num 和 $next_seq_num$ 之间的距离，窗口变大。



snd_wnd_pointer	snd_wnd_left+cwnd	snd_wnd_left+rcv_wnd	snd_wnd_pointer-left
1769176648	1769179567	1769176647	1
1769176648	1769179568	1769242183	0
1769176648	1769179568	1769242183	0
1769178048	1769179568	1769242183	1400
1769179496	1769179568	1769242183	2848
1769179496	1769183876	1769245031	0

接收窗口示意图

