

TCP/IP 通訊協定及應用

Spring 2002

中央大學 吳曉光博士

<http://wmlab.csie.ncu.edu.tw/course/tcp>

We
provide
無線網路多媒體實驗室
Wireless
Wireless Network & Multimedia Laboratory
Solution

Chapter 22: TCP Persist Timer

Introduction

- ◆ Why we use persist timer:
 - If a acknowledgment is lost, we could end up with both sides waiting for the other
 - ◆ To prevent deadlock from occurring the sender uses a persist timer
 - ◆ These segments from the sender are called window probes

- ◆ An Example:
 - We'll invoke the server as
 - ◆ `svr4 % sock -i -s -P100000 5555`
 - The server sleep for 100,000 seconds(27.8 hours) before reading from the network
 - The client on host bsdi and performs 1024-byte writes to port 5555 on the server

An Example

Establishment from the output:

```

1      0.0      bsdi.1027 > svr4.5555: P 1:1025(1024) ack 1 win 4096
2      0.191961 ( 0.1920) svr4.5555 > bsdi.1027: . ack 1025 win 4096
3      0.196950 ( 0.0050) bsdi.1027 > svr4.5555: . 1025:2049(1024) ack 1 win 4096
4      0.200340 ( 0.0034) bsdi.1027 > svr4.5555: . 2049:3073(1024) ack 1 win 4096
5      0.207506 ( 0.0072) svr4.5555 > bsdi.1027: . ack 3073 win 4096
6      0.212676 ( 0.0052) bsdi.1027 > svr4.5555: . 3073:4097(1024) ack 1 win 4096
7      0.216113 ( 0.0034) bsdi.1027 > svr4.5555: P 4097:5121(1024) ack 1 win 4096
8      0.219997 ( 0.0039) bsdi.1027 > svr4.5555: P 5121:6145(1024) ack 1 win 4096
9      0.227882 ( 0.0079) svr4.5555 > bsdi.1027: . ack 5121 win 4096
10     0.233012 ( 0.0051) bsdi.1027 > svr4.5555: P 6145:7169(1024) ack 1 win 4096
11     0.237014 ( 0.0040) bsdi.1027 > svr4.5555: P 7169:8193(1024) ack 1 win 4096
12     0.240961 ( 0.0039) bsdi.1027 > svr4.5555: P 8193:9217(1024) ack 1 win 4096
13     0.402143 ( 0.1612) svr4.5555 > bsdi.1027: . ack 9217 win 0
14     5.351561 ( 4.9494) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
15     5.355571 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
16    10.351714 ( 4.9961) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
17    10.355670 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
18    16.351881 ( 5.9962) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
19    16.355849 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
20    28.352213 (11.9964) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
21    28.356178 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
22    52.352874 (23.9967) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
23    52.356839 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
24   100.354224 (47.9974) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
25   100.358207 ( 0.0040) svr4.5555 > bsdi.1027: . ack 9217 win 0
26   160.355914 (59.9977) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
27   160.359835 ( 0.0039) svr4.5555 > bsdi.1027: . ack 9217 win 0
28   220.357575 (59.9977) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
29   220.361668 ( 0.0041) svr4.5555 > bsdi.1027: . ack 9217 win 0
30   280.359254 (59.9976) bsdi.1027 > svr4.5555: . 9217:9218(1) ack 1 win 4096
31   280.363315 ( 0.0041) svr4.5555 > bsdi.1027: . ack 9217 win 0

```

Figure 22.1 Example of persist timer probing a zero-sized window.

An Example

- ◆ About persist timer:
 - It always bounded between 5 and 60 seconds
 - The persist state is different from the retransmission timeout is that TCP never gives up sending window probes
- ◆ What is Silly Window Syndrome
 - It can be caused by the receiver can advertise small windows and the sender can transmit small amounts of data
- ◆ Correct avoidance of the silly window syndrome
 - 1.The receiver must not advertise small windows
 - 2.Sender is not transmitting unless one of conditions is true:
 - ◆ a.a full-sized segment can be sent
 - ◆ b.we can send at least one-half of the maximum sized window ever advertised
 - ◆ c.we can send everything we have and either we are not expecting an ACK

Silly Window Syndrome

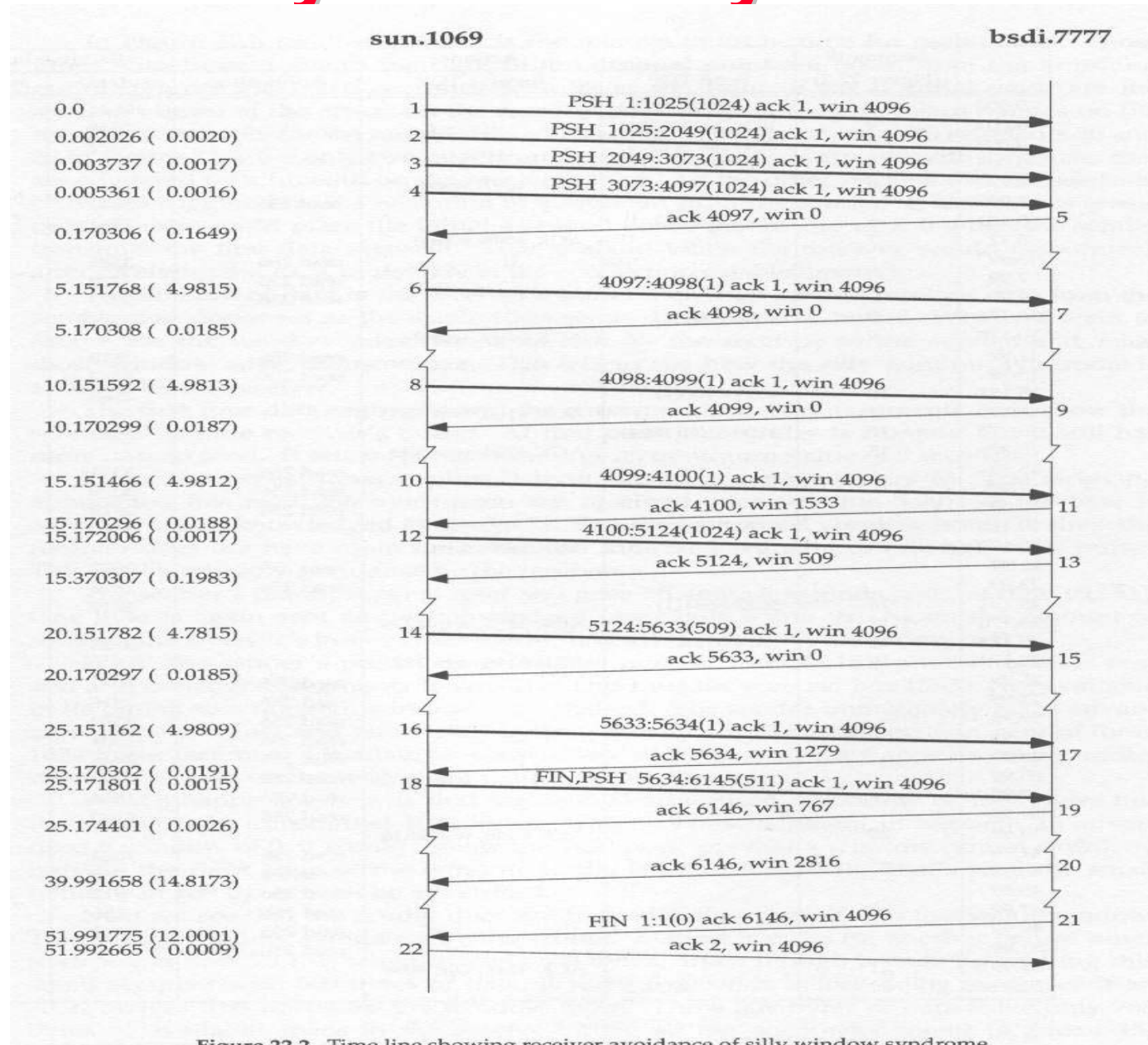


Figure 22.2 Time line showing receiver avoidance of silly window syndrome.

Silly Window Syndrome

Time	Segment# (Figure 22.2)	Action			Receiver buffer	
		Send TCP	Receive TCP	Application	data	available
0.000	1	1:1025(1024)			1024	3072
0.002	2	1025:2049(1024)			2048	2048
0.003	3	2049:3073(1024)			3072	1024
0.005	4	3073:4097(1024)			4096	0
0.170	5		ACK 4097, win 0			
3.99				read 256	3840	256
5.151	6	4097:4098(1)			3841	255
5.17	7		ACK 4098, win 0			
5.99				read 256	3585	511
7.99				read 256	3329	767
9.99				read 256	3073	1023
10.151	8	4098:4099(1)			3074	1022
10.170	9		ACK 4099, win 0			
11.99				read 256	2818	1278
13.99				read 256	2562	1534
15.151	10	4099:4100(1)			2563	1533
15.170	11		ACK 4100, win 1533			
15.172	12	4100:5124(1024)			3587	509
15.370	13		ACK 5124, win 509			
15.99				read 256	3331	765
17.99				read 256	3075	1021
19.99				read 256	2819	1277
20.151	14	5124:5633(509)			3328	768
20.170	15		ACK 5633, win 0			
21.99				read 256	3072	1024
23.99				read 256	2816	1280
25.151	16	5633:5634(1)			2817	1279
25.170	17		ACK 5634, win 1279			
25.171	18	5634:6145(511)			3328	768
25.174	19		ACK 6146, win 767			
25.99				read 256	3072	1024
27.99				read 256	2816	1280
29.99				read 256	2560	1536
31.99				read 256	2304	1792
33.99				read 256	2048	2048
35.99				read 256	1792	2304
37.99				read 256	1536	2560
39.99				read 256	1280	2816
39.99	20		ACK 6146, win 2816			
41.99				read 256	1024	3072
43.99				read 256	768	3328
45.99				read 256	512	3584
47.99				read 256	256	3840
49.99				read 256	0	4096
51.99				read 256 (EOF)	0	4096
51.991	21		ACK 6146, win 4096			
51.992	22	ACK 2				

Figure 22.3 Sequence of events for receiver avoidance of the silly window syndrome.

Summary

- ◆ TCP's persist timer is set by one end of a connection but has been stopped because the other end has advertised a zero-sized window
- ◆ TCP's avoidance of the silly window syndrome is to prevent TCP from advertising small windows or from sending small segments