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

Chapter 23:
TCP Keepalive Timer

Introduction

- The keepalive timer provides the capability to let a server wants to know if the client's host has either crashed and is down , or crashed and rebooted.
  - The keepalive is intended for server applications that might tie up resources on behalf of a client , and want to know if the client host crashes.
  - The keepalive is intended to detect those half-open connection from the the server side.

Introduction

- Keepalive are not part of the TCP specification. The Host Requirements RFC provides three reasons not to use them.
  - They can cause perfectly good connections to be dropped during transient failures.
  - They consume unnecessary bandwidth .
  - They cost money on an internet that charges by the packet.

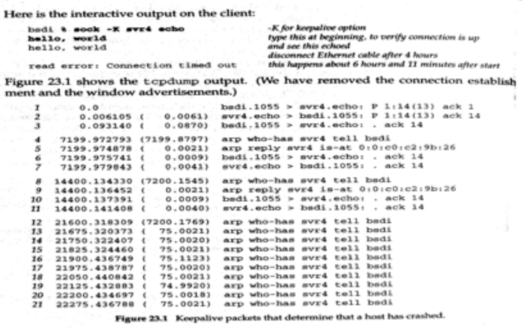
Description

- The end that enables the keepalive option is server , and other is the client .
- If there is no activity on a given connection for 2 hours , the server sends a probe segments to the client.
- The client host must be one of four states.
  - The client host is still up and running and reachable from the server .
  - The client's host has crashed and is either down or in the process of rebooting.
  - The client's host has crashed and rebooted .
  - The client's host is up and running , but unreachable from the server.

Keepalive Example

- Other End Crashes
  - establish a connection between a client ***bsdi*** and the standard echo server on the host ***svr4***.
  - Verify that data can go across the connection.
  - Watch the client's TCP send keepalive packets every 2 hours and see them acknowledged by the server's TCP.
  - Disconnect the Ethernet cable from the server, and leave it off until the example is complete.
  - The client send 10 keepalive probes, 750 seconds apart before declaring the connection dead.

Keepalive Example



Here is the interactive output on the client:

```
bsdi % sock -K svr4 echo                  -K for keepalive option
hello, world                              type this at beginning, to verify connection is up
hello, world                              and see this echoed
                                          disconnect Ethernet cable after 4 hours
read error: Connection timed out          this happens about 6 hours and 11 minutes after start
```

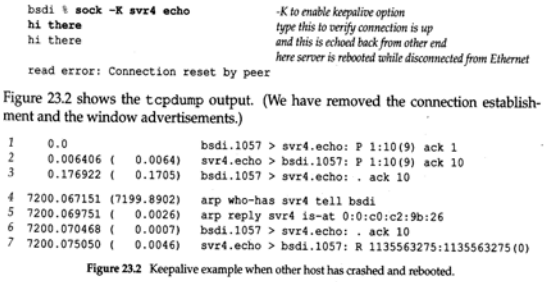
Figure 23.1 shows the tcpdump output. (We have removed the connection establishment and the window advertisements.)

```
 1      0.0                 bsdi.1055 > svr4.echo: P 1:14(13) ack 1
 2      0.006105 (   0.0061) svr4.echo > bsdi.1055: P 1:14(13) ack 14
 3      0.093140 (   0.0870) bsdi.1055 > svr4.echo: . ack 14

 4   7199.972793 (7199.8797) arp who-has svr4 tell bsdi
 5   7199.974878 (   0.0021) arp reply svr4 is-at 0:0:c0:c2:9b:26
 6   7199.975741 (   0.0009) bsdi.1055 > svr4.echo: . ack 14
 7   7199.979843 (   0.0041) svr4.echo > bsdi.1055: . ack 14

 8  14400.134330 (7200.1545) arp who-has svr4 tell bsdi
 9  14400.136452 (   0.0021) arp reply svr4 is-at 0:0:c0:c2:9b:26
10  14400.137391 (   0.0009) bsdi.1055 > svr4.echo: . ack 14
11  14400.141408 (   0.0040) svr4.echo > bsdi.1055: . ack 14

12  21600.318309 (7200.1769) arp who-has svr4 tell bsdi
13  21675.320373 (  75.0021) arp who-has svr4 tell bsdi
14  21750.322407 (  75.0020) arp who-has svr4 tell bsdi
15  21825.324460 (  75.0021) arp who-has svr4 tell bsdi
16  21900.436749 (  75.1123) arp who-has svr4 tell bsdi
17  21975.438787 (  75.0020) arp who-has svr4 tell bsdi
18  22050.440842 (  75.0021) arp who-has svr4 tell bsdi
19  22125.432883 (  74.9920) arp who-has svr4 tell bsdi
20  22200.434697 (  75.0018) arp who-has svr4 tell bsdi
21  22275.436788 (  75.0021) arp who-has svr4 tell bsdi
```

Figure 23.1 Keepalive packets that determine that a host has crashed.

Keepalive Example

- Other end crashes and reboots



```
bsdi % sock -K svr4 echo                  -K to enable keepalive option
hi there                                  type this to verify connection is up
hi there                                  and this is echoed back from other end
                                          here server is rebooted while disconnected from Ethernet
read error: Connection reset by peer
```

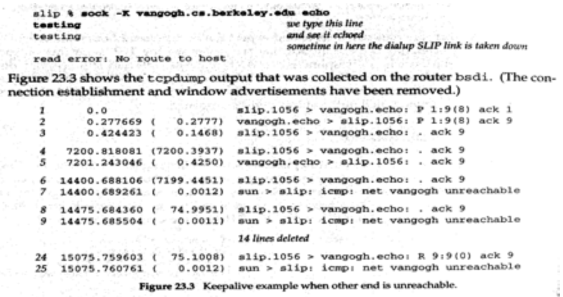
Figure 23.2 shows the tcpdump output. (We have removed the connection establishment and the window advertisements.)

```
1    0.0                  bsdi.1057 > svr4.echo: P 1:10(9) ack 1
2    0.006406 (   0.0064) svr4.echo > bsdi.1057: P 1:10(9) ack 10
3    0.176922 (   0.1705) bsdi.1057 > svr4.echo: . ack 10

4 7200.067151 (7199.8902) arp who-has svr4 tell bsdi
5 7200.069751 (   0.0026) arp reply svr4 is-at 0:0:c0:c2:9b:26
6 7200.070468 (   0.0007) bsdi.1057 > svr4.echo: . ack 10
7 7200.075050 (   0.0046) svr4.echo > bsdi.1057: R 1135563275:1135563275(0)
```

Figure 23.2 Keepalive example when other host has crashed and rebooted.

Keepalive example

- Other end is unreachable



```
slip % sock -K vangogh.cs.berkeley.edu echo
testing                                   we type this line
testing                                   and see it echoed
                                          sometime in here the dialup SLIP link is taken down
read error: No route to host
```

Figure 23.3 shows the tcpdump output that was collected on the router bsdi. (The connection establishment and window advertisements have been removed.)

```
 1      0.0                 slip.1056 > vangogh.echo: P 1:9(8) ack 1
 2      0.277669 (   0.2777) vangogh.echo > slip.1056: P 1:9(8) ack 9
 3      0.424423 (   0.1468) slip.1056 > vangogh.echo: . ack 9

 4   7200.818081 (7200.3937) slip.1056 > vangogh.echo: . ack 9
 5   7201.243046 (   0.4250) vangogh.echo > slip.1056: . ack 9

 6  14400.688106 (7199.4451) slip.1056 > vangogh.echo: . ack 9
 7  14400.689261 (   0.0012) sun > slip: icmp: net vangogh unreachable

 8  14475.684360 (  74.9951) slip.1056 > vangogh.echo: . ack 9
 9  14475.685504 (   0.0011) sun > slip: icmp: net vangogh unreachable
                             14 lines deleted
24  15075.759603 (  75.1008) slip.1056 > vangogh.echo: R 9:9(0) ack 9
25  15075.760761 (   0.0012) sun > slip: icmp: net vangogh unreachable
```

Figure 23.3 Keepalive example when other end is unreachable.

Summary

- The keepalive feature is controversial .
- Sending a probe packet across a connection after the connection has been idle for 2 hours, four different scenarios can occur :
  - the other end is till there.
  - The other end has crashed.
  - The other end has crashed and reboot.
  - The other end is currently unreachable.