

# Re: NAT and keepaliveopen connection over TCP

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.networks/2006-11/msg00195>

---

- *From:* "semedao" <[semedao@xxxxxxxxxxxxxxxxxxxx](mailto:semedao@xxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 15 Nov 2006 10:41:47 +0200
- 

I look at

<http://msdn2.microsoft.com/en-us/system.net.sockets.iocontrolcode.aspx>

and they point to the RFC 1122 section 4.2.3.6:

4.2.3.6 TCP Keep-Alives

Implementors MAY include "keep-alives" in their TCP implementations, although this practice is not universally accepted. If keep-alives are included, the application MUST be able to turn them on or off for each TCP connection, and they MUST default to off.

Keep-alive packets MUST only be sent when no data or acknowledgement packets have been received for the connection within an interval. This interval MUST be configurable and MUST default to no less than two hours.

It is extremely important to remember that ACK segments that contain no data are not reliably transmitted by TCP.

Consequently, if a keep-alive mechanism is implemented it MUST NOT interpret failure to respond to any specific probe as a dead connection.

Now, Do you think it's a solution to use the IOControlCode...

It will not make the socket crazy ? :)

and how I should define it to send keep alive every 30 seconds or every other interval ? (c#)

maybe the best thing will be to in some way know when we are behind some NAT like this..and only then to send every X seconds

when the other nat's can use the regular 2 hours interval ?

thanks again

"Alexander Nickolov" <[agnickolov@xxxxxxxx](mailto:agnickolov@xxxxxxxx)> wrote in message <news:u0T6wnCCHHA.4428@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

A more primitive NAT could do that, sure. Your connection is busted at that point. The client will realize it after it's sent back an RST. The server will realize it much later when

Re: NAT and keepaliveopen connection over TCP

the TCP keep-alive kicks in (10 minutes of inactivity). The basic problem is of course inactivity on the connection. There's no minimum set time how long a NAT router should maintain its bindings. Thus if you send data after the binding has expired, the NAT router simply creates a new binding naturally using a new public port number. I said more primitive NAT routers, because most NAT routers actually inspect the TCP packets a bit more and only open new bindings for TCP packets containing the SYN flag. Other TCP packets would cause the NAT router to drop the packet and return an RST packet back to your client.

The only solution to your problem is if you send keep-alive messages yourself as part of your protocol and actively time-out inactive connections at the server. While there's no minimum inactivity time for NAT bindings to expire, a good number to pick would be 30 seconds to 1 minute between keep-alives. Shorter times means increase in useless traffic, longer times increase the risk of NAT bindings expiration. YMMV...

--  
=====  
Alexander Nickolov  
Microsoft MVP [VC], MCSO  
email: agnickolov@xxxxxxx  
MVP VC FAQ: <http://www.mvps.org/vcfaq>  
=====

"semedao" <semedao@xxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:OqBpfTCCHHA.4844@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:OqBpfTCCHHA.4844@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)  
Hi,

I have a TCP server that listening to tcp clients, this server can accept incoming tcp requests.

Some of the clients are behind NAT.

The client makes connection, and then I set keepalive on both sides (server & client)

in some Nat's it's work fine , and in others the client suddenly after work correctly send the packets with other port (external – Nat port) to the server , even if I use the same already opened socket !

For ex.

Client A (192.168.1.1) is behind NAT B (60.78.95.144) make connection to Server S (87.170.65.132) on that listening on port 1000.

The NAT will change the port number from 1000 to 2000

## Re: NAT and keepaliveopen connection over TCP

The connection established.

Then

When the connection is still established the client try to send let say 30 bytes to the server

In the server we have connection to 60.78.95.144:2000 and we try to read from it.

But the packet from the NAT will come from 60.78.95.144:3000

What cause it?

There is some specific Nat that make it?

How can I identify that Nat will act like this (in the program c#)

How to correct it?

Maybe I should avoid the keepalive and use my "keep alive" by sending packets to the server every X interval? (And if yes, how to know what is the interval)

Or maybe the server should send to the client?

Thanks