It's a common networking problem, if there are routers at BOTH ends of the connection then TPC/IP can't route traffic properly. That's why if you host any server type applications (Ren FDS, FTP server etc etc) on a machine behind a router a lot of people won't be able to connect to them.

I can't remember exactly what the technical term for it is, but I read it somewhere when I was reading some info on windows server 2003 which uses some fancy software to get around this...

EDIT: this might not mean much to you but this is what I was talking about

Quote:NAT Traversal
First, XP introduced NAT Traversal. For those who don't know what that is, NAT Traversal tries to solve the problem of "how do I communicate from inside one NAT network to another?"

More specifically: suppose you've got a cable modem or DSL connection with a connection sharing device of some kind, like a DSL router. The DSL router has two IP addresses. First, there's the honest-to-God, fully routable IP address that it got from your Internet provider, connected to the DSL or cable modem connection. Then there's the connection to a switch that you've got all of your internal machines connected to—the old Windows 9x boxes, NT machines, 2000 systems, Macintoshes, or whatever. The DSL router's job is to share the one "legal" Internet address among several devices. But every device needs a unique IP address. Lots of devices, but just one IP address—what to do?

As you may know, DSL routers solve this problem by giving all of the internal systems—those Windows, NT, 2000, and Mac machines—IP addresses from a block of addresses set aside to be nonroutable. Anyone can use them.

There are several of these nonroutable blocks, but most DSL routers seem to use the 192.168.1.x or 192.168.0.x subnets. The DSL routers then use something called network address translation or, more correctly, port address translation (again, see Chapter 6 if this isn't familiar) to share the one routable address with all of the internal systems. How it does it is pretty simple: whenever an internal system wants to access the Internet, perhaps to browse some Web site, then that system just says to the DSL router, "Please forward this request to Internet address so-and-so," as routers normally do. But the DSL router knows perfectly well that it can't do that: if it says to the Internet, "Hey, someone at 192.168.1.3 has a request," then the first Internet router to see the message will simply refuse to route it, as the address is in a range of addresses that are, by definition, NONroutable. So the DSL router doesn't say "192.168.1.3 wants something"; instead, the DSL router substitutes its routable address. Then, when the answer to 192.168.1.3's question comes back, the DSL router remembers which machine asked the question in the first place and routes the answer to 192.168.1.3. The result is that to the general Internet, that DSL router sure seems like a demanding system, when in fact it is simply busy because it is impersonating a bunch of systems.

In any case, notice that it's possible for an internal system (one with one of those 192.168.x.x addresses) to initiate a communication with a device on the public, routable Internet, but it's NOT possible for a device on the public, routable Internet to initiate a conversation with an internal 192.168.x.x system.

Here, then, is the problem. Suppose I'm sitting at a Windows 2000 Pro box in my home that has a 192.168.x.x address, accessing the Internet via my DSL router or cable modem sharing device. You're sitting in your house, also using some kind of DSL router or cable modem sharing device to access the Internet. We meet on-line and decide to play some networkable game and start to set up our connection. One of us acts as the server and one as the client. The client then initiates communication with the server. That's where the problem appears. I could initiate a communication to a routable address, or YOU could initiate a communication to a routable address, but neither of us has a routable address… and so we can't communicate.