A Note on Connecting a Web Server to the Internet

By Bob Walsh

Geist Technology Inc

January 16, 2009

A Note on Connecting a Web Server to the Internet

There are some important issues that must be understood in order to properly connect one of Geist Technology Web Servers to the Internet. Because the device is a "server" rather than a "client", connecting it is not as simple as connecting a laptop to the web. The problem arises because a web server has to have an IP address that can be reached from other places on the Internet. The purpose of this paper is to make you aware of the issues this raises so you can have a solution by the time your server is ready to be installed.

This issue comes up only if you intend to connect your server to the Public Internet. If you are installing it on some private network and only intend to access it from that private network you won't have this problem.

This note is aimed at people who have little or no knowledge of the Internet. I have tried to limit the technical details as much as possible, but some tech content is unavoidable. If you have a strong background in how the Internet works you probably know most of this already.

Servers and Clients on the Web

What does it mean to have an IP address that can be reached from elsewhere on the Internet? First, some background on the World Wide Web: The Web is built on what is called a "client/server" model. Servers sit and wait for clients to request services. When a client requests service, a server answers the request. Servers don't do anything until some client asks them to. When a user at a PC running a Web Browser (such as Firefox, Internet Explorer, or Safari) wants to see a web page, the browser is acting as a client. The browser sends a message to the appropriate server asking for that page. The server responds by sending the page data. The user may follow a "link" on the page, which results in a request for a different page. The link might reference a page on another server, in which case the request goes to that server. Some pages ask the user to enter some information, and the browser sends that information back to the server.

The key issue is this: how does the user's browser know how to reach the server? The answer is perhaps a little more complicated than you might expect.

Internet Addressing

First, we have to discuss how addressing works on the internet. Any device that is connected to the Public Internet has to have an Internet Protocol Address, or an "IP" address. An IP address is really just a number in the range 0 to 4,294,967,295. For reasons beyond the scope of this discussion, an IP address is usually written as four numbers separated by periods, like "192.168.140.29" or "67.79.205.85". Every device on the Internet must have an IP address, and each address must be unique.

Where do these addresses come from? At the highest level addresses are assigned by an outfit called IANA. IANA assigns addresses in large blocks to government agencies, universities, major corporations, and large internet service providers (ISPs) who in turn assign smaller blocks to smaller providers who eventually assign individual addresses to their customers.

Routing

Now a note on what is called "routing"; that is how does a message from one device get passed through the Internet to its destination? What we call the" Internet" is really a loosely organized collection of local networks connected to each other using special purpose computers called "routers". Routers pass messages between networks. At the edges of the internet some clients and maybe a server or two are directly connected to each other in local networks called "subnets". There are thousands or perhaps millions of these subnets on the Internet. Each subnet is connected to at least one router that connects it to the rest of the Internet. Routers in turn are connected to other routers which are connected to even more routers. The routers talk to each other to build "routing tables" which they use to find the most efficient route through the Internet to a given subnet. When a message comes to a router, the router examines the destination address, more specifically the first two or three of the four numbers in the address, which is the address of the subnet that the destination device belongs to. (The last number or two is the address of that individual device in the subnet). The router looks at its routing tables and consults some rules to decide which router it should send the message to next. A given message might pass through several routers on the way to its destination. (The details of how this works is very complicated and far above the level of this discussion.) Finally it arrives at the last router in the chain and is passed to the destination device.

Domain Name Service

When you sit down at your browser you don't usually enter an IP address to find a web server. You usually enter something called a Uniform Resource Locator ("URL") such as http://www.GeistTek.com. How does the URL get translated to an IP address? A service called "Domain Name System" ("DNS") does that. DNS is actually a group of special servers on the internet. To access the Internet by URLs instead of IP addresses your PC has to know the IP address of at least one DNS server. (Your ISP usually sets that up when you set up your account) When you enter a URL, your PC sends a DNS request and the server responds with the matching IP address. The DNS system has to know the IP address corresponding to every URL on the Internet. When a new URL is created someone has to tell DNS what that IP address is. If you enter a URL that is not in DNS into your browser, you get an error message.

Static IP Addresses

So what's the problem? If you want to connect up your new server to the Internet, you just get an IP address assigned to it somehow and you're done, right? An address that is dedicated to an individual server all of the time is what is called a "static" IP address. Commercial servers like www.GeistTek.com pay for static IP addresses because we handle a large volume of requests. Unfortunately it might be very expensive or even impossible at the location where you want to install your server.

So why can't everyone have a static IP address? The problem, believe it or not, is that there is a shortage of addresses. At this beginning of this paper I implied that there are more than four billion possible addresses. That's a lot of addresses! Unfortunately, they're really not all useable. When the Internet was first conceived, computers were big and expensive and there weren't many of them. The developers never dreamed that there would be as many devices on the net as there are today. Decisions were made that affected how many of those addresses are really useable, and the result is that there is a shortage today. The most common solution is to use what are called "dynamic IP addresses". If you buy Internet service from the phone company, cable company, or a traditional dial up service provider you will usually get a dynamic IP address. This works fine as long as your machine acts as a client, as described above. Static addresses are reserved for servers. Since most service providers only have a limited number of static addresses allocated to them, they charge a higher price to a user who needs a static address, and they may have no more available for new users.

Dynamic IP Addresses

Dynamic IP addresses allow an ISP to have more subscribers than the ISP has IP addresses, thus alleviating the shortage. The catch is that all the users can't be connected all at once, but that's usually okay. When a user starts to access the Internet, a server at the ISP (yep, another server) "leases", or temporarily assigns an IP address to him. This works fine since his PC is acting as a client rather than a server. The browser requests service from various web servers, and his requests include his return address, so the responses reach him. If the user hangs up, or simply doesn't access the Net for some fixed time (30 minutes or an hour typically) the lease expires and the IP address is available to be leased to some other user. When the first user reconnects he gets another address, which is probably not the same as the one he had the last time. This, finally, is the problem with trying to connect a server the same way you connect a client. There has to be some way that other clients out on the Net can know the IP address of your server, and if your server has a dynamic address they can't know it because it may change often. So this is the problem we have to solve.

One Solution: A Static IP Address

There are at least two solutions to this problem. The simplest, as noted above, is to buy a Static IP address from some provider. You set up your server with this address and register your domain name with DNS. Then your clients can find you in the time honored fashion. Unfortunately a static IP address may not be available where your need to install your server, or the provider might want to charge a large monthly fee for one.

Another Solution: Dynamic DNS

Another solution is to use a service called Dynamic DNS ("DDNS"). Briefly, Dynamic DNS is a service that solves the problem of running a server with a dynamic address. It does this by using a special program on the server that updates Dynamic DNS whenever the server's address changes. Several companies offer free DDNS service, including www.dyndns.com. A good overview of Dynamic DNS can be found at http://www.technopagan.org/dynamic.

For more information on Internet routers, try this link:

http://computer.howstuffworks.com/computer-networking-channel.htm

For information on Dynamic DNS see:

http://en.wikipedia.org/wiki/Dynamic_DNS_Update

1 December 2008

Bob Walsh:

Senior Applications Engineer

Geist Technology Inc

bwalsh@GeistTek.com