Global Information Assurance Certification Paper

Title: TCP Tunnels, where University and Security Policy beaks down.
By Jonathan Sloop

Problem: TCP Tunnels are circumventing University policy restrictions. Port restrictions on our firewall edge device had been blocking outbound Napster and similar file sharing services. Now the word on TCP Tunneling is spreading among users and is providing them with unauthorized Internet file sharing, and potentially giving hackers a backdoor to our private network.

Situation: Our University does not have an official written policy against Napster or Napster like Internet file sharing programs, however we have not opened the associated ports on our firewall to allow default access. The reason for this is fear. Fear of the uproar from the students if we officially blocked Napster. Fear that if Napster were allowed to run freely on our campus the network would grind to a halt. Collision domain would collide and our Internet connection would seize. This University has no desire to become a sensor on how the Internet is used, but at the same time we feel some responsibility to provide reliable consistent access for students educational endeavors.

Background Information: For the past several years the University has provided desktop computers in every dorm room. The total number of University owned PCs on campus is quickly approaching 4000. There are no restrictions on students adding their own PCs to the network. Our Internet connection is currently a throttled 10Mb/s on an OC3 line. This can be quickly opened to 45Mb/s if there is justification and someone is willing to pay the additional tariffs. Several years ago we were a 12 class C network desperately trying to figure out how to keep 4000 computers on the Internet with only 3000 or so IP address. We also had several unauthorized servers on campus providing Web, Ftp, Telnet, etc. to the Internet. There were even cases of former students leaving connected servers running commercial web applications in dorm rooms of friends. Why not, the University was proving free Internet access. Even with a policy against student servers no one was attempting to enforce it. One way identified to help curb these problems was to install a firewall.

Firewalls and Universities are kind of like trying to mix oil and water. The University's faculty will maintain that open access is necessary for research and educational endeavors. The student's will insist it is their constitutional right under the first amendment to do what ever they want. In the end networking people have an uphill battle. Convincing all interested parties that it's necessary to spend tens of thousands of dollars funding a project that conflicts with the faculty and student's opinions isn't easy. For this University it took outside influences to finally make a commitment to a firewall project.

Along came the FBI, remember those unauthorized servers. Apparently some of those servers were used to bounce attacks at some high profile sites out on the Internet. The FBI promptly introduced the terms Firewall and Network Security to the University's decision-makers. Still nothing was done, no budgets were created, almost no reaction at all, but the idea was rolled around. This new concept called Network Security might be worth looking into, sometime, maybe later? Then came the Audit, the auditors took a look around and said, how are you protecting your mission critical data, students grades, protecting everyone's privacy? How are you keeping the bad people out there on the Internet from connecting to your data systems? How are you keeping students form accessing sensitive data? Where are your firewalls?

The Networking Staff had already done extensive research into firewalls and network security in general. We knew that we could use NAT (Network Address Translation) sometimes referred to as IP masquerading, to solve our IP shortage.(1) We would simply use a single external IP address and a class A private internal address scheme. We would use the private or non-routable class A network 10.0.0.0 with over 16 million IP addresses on our private network. Non-routable IP addresses are covered under RFC 1597.(2) With more than enough IP addresses to go around and the added security of a non-Internet routable address set to boot. The only down side to NAT would prove to be a requirement from our state funded ISP to log every connection to the Internet, ouch! They understandably want to retain the ability to collect forensic evidence in the event some unscrupulous University user were to attack an Internet site without proper authorization. Students are often very surprised to hear that the first amendment does not protect them from their own malicious activities over the Internet. Unfortunately our connection logs are one to two gigabytes almost daily. This means to keep logs for 30 days we need somewhere between 40 and 60

gigabytes of active storage. Permanent storage is an even bigger problem, maybe no one will ask for last month's logs.

Installing a firewall would help us prevent External Hosts (People on the Internet) from connecting to servers setup by students on the private network. It should be noted that we did have an IT staff member setup a Linux box that was hacked from the Internet, then used to leapfrog an attack to another high profile web site(so the students aren't always to blame). That site graciously called the FBI on us, that was our second visit, actually I think they just called this time. By not allowing external hosts to initiate connections to computers on the private network, we believed we would be protecting our Internal users form the big bad Internet, and preventing unauthorized use of our Internet connection. Building on the concept of a Layered Defense, (for those of us in plausible denial, "a defense in depth") there are two mechanisms that we believed would prevent an External to Internal connection. The first being NAT, it is literally impossible (assuming all routers in the hop across the Internet are configured correctly) to connect to a computer on our Private network via its 10.0.0.0 IP address. Even if an intruder somehow knew the IP address of one of the PCs on our private network the first router on the Internet the Hacker hit would drop the packet. With a NAT/Firewall, from the External network's point of view all 4000 computers appear to have the same IP address. So if a Hacker were to attempt to connect to a PC on the Private network using the External IP address of the NAT/Firewall they would simply get dropped. The NAT/Firewall would not know which of the 4000 Private PCs it was supposed to forward the packets to, so it gets rid of it. The second is the rule base or policy list on the Firewall. If the rules say don't allow External hosts to connect to Private hosts it won't let them. TCP Tunneling has proven to defeat both of our Layered Defenses.

For the most part when you install a proxy type firewall its default configuration has all TCP and UDP ports blocked. I have often heard this referred to as, "What's not explicitly allowed is denied." All of the software firewall products I have used followed this simple rule. Our University policy for port opening reads:

> **External and Internal Private Networks:**
>
> Any identified and approved ports will be opened for outgoing IP TCP/UDP traffic, Internal Private host requests will be allowed to pass through the firewall, their immediate response will be returned to the Internal Private requesting host. All Internal Private hosts will appear to have the IP address of the external firewall network interface as seen by the External network, this is referred to as Network Address Translation (NAT). All incoming traffic from the External network that is not in response to an Internal Private host request will be blocked.
>
> **External and Public Networks:**
>
> Requests from External network hosts will be allowed to access public services on public servers through associated ports (i.e. web services through port 80 TCP).
>
> The procedure for requesting that a port or service be allowed to pass through the firewall from the Internal Private or External network is to send an email to the Firewall Administrator at administrator@OMITED.edu. The request will be researched then submitted to the Network Security Council for review and approval.
>
> Note: The firewall is designed to protect Internal Private users from IP based attacks originating from the External network and therefor regrettably limits some Internet services.

TCP Tunneling: There has been much said about the legal ramifications of passing copyrighted material within Napster's community and its associated software. Eventually the courts will decide whether or not Napster will be allowed to conduct business as it has from its inception. The Internet has long been an

effective means of stealing intellectual property. Crackers and Hackers have had a long history of disregarding copyright laws, Napster is seams is just a means to that end. I think in the end it will come down to whether Napster knowingly allows its users to break copyright laws, and I don't see how they can't. The University quite frankly, just wants to protect its bandwidth, therefore no Napster.

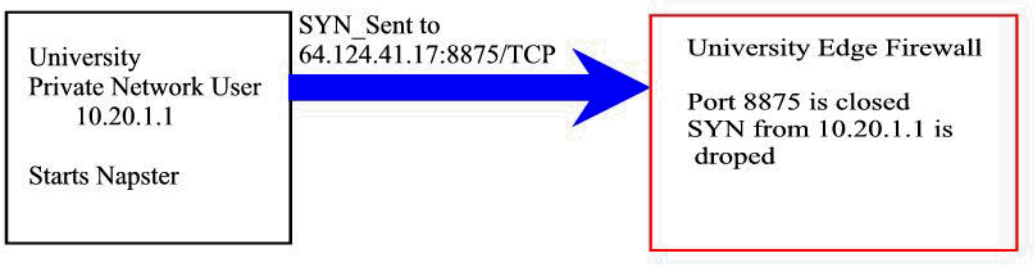
What is Napster up to? From behind our firewall, from a technical perspective, not much, that is until you tunnel. I used the command 'netstat –a –n 1' at a DOS prompt to watch what connections and ports were active during a Napster session. Netstat with a '–a' will display the active connections, '-n' will keep the application from trying to resolve Foreign (IP) Address to a Domain Name, and the '1' is the number of seconds till the data is refreshed. Napster v2.0 BATA 7 first makes a connection to 64.124.41.19:8875/TCP SYN_SENT then 64.124.41.17:8875/TCP SYN_SENT.(3) Because our firewall does not have a port open for 8875 TCP the first SYN to 64.124.41.19 gets no response. So the Napster program sends a second SYN to 64.124.41.17 on the same port, this also gets to the firewall and is dropped. Napster apparently gives up and prompts the user with an, "Unable to Connect to Server!" message. Note: I have found that Napster has several IP addresses within the 64.124.41.0 address space that they use for connections, so if you try this you may get slightly different results. For a longtime this has been sufficient to discourage users at our University from using Napster and freeing us from potential bandwidth problems not to mention legal issues. We are sort of taking the easy way out. By not having an official policy against Napster we haven't come under any scrutiny for censuring web sites, and Napster did not work.

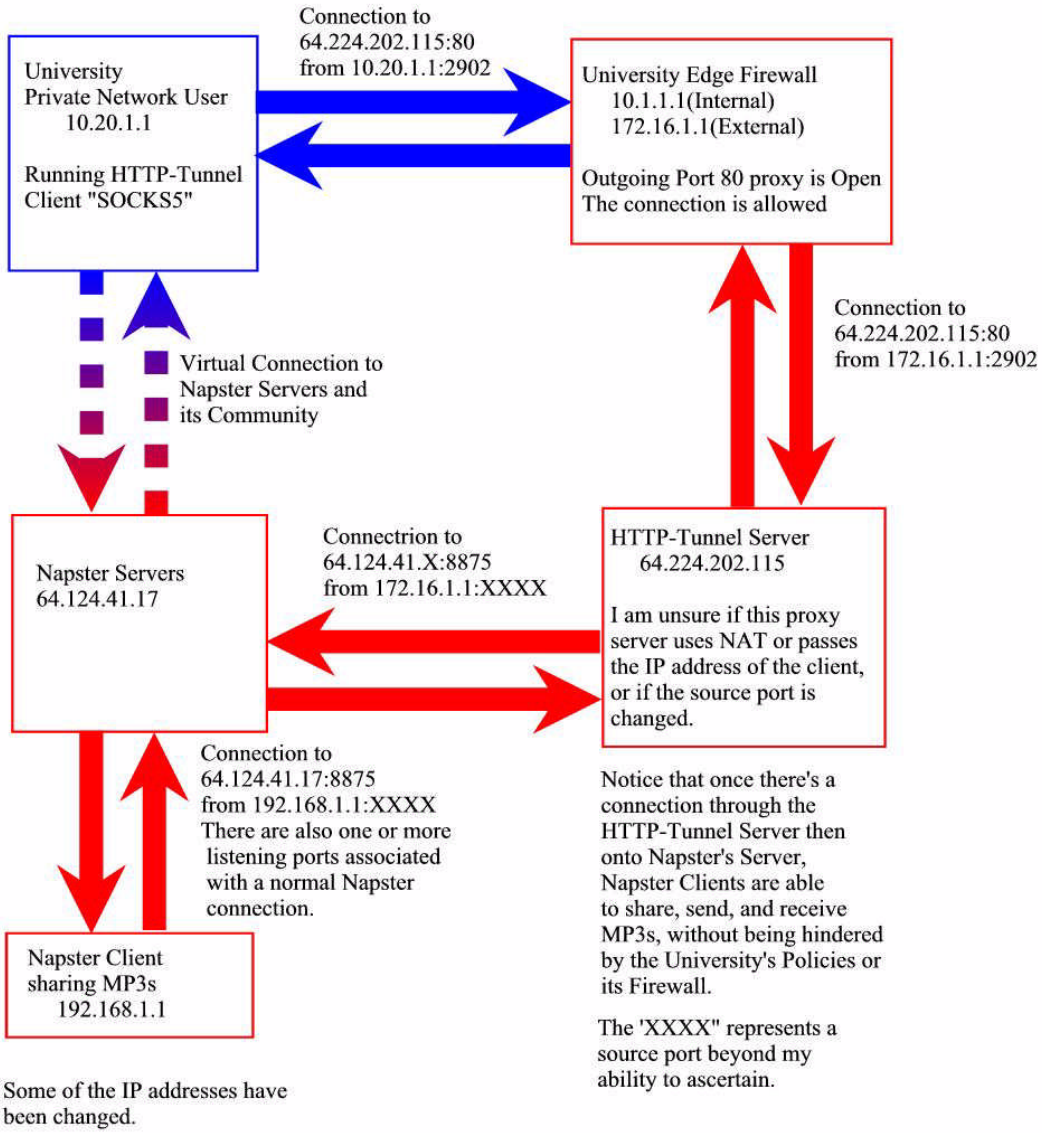
Rumor has it that our students have figured out a way around the closed port 8875 policy and are now readily using Napster. It looks like the majority of our growing Napster users are using a TCP Tunnel to circumvent the blocked 8875 port. I ran the 'netstat –a –n 1' command at a DOS prompt then started the HTTP-Tunnel Client v2.3.1587 (Beta), I downloaded from their site.(4) It clams to be free but never the less relentlessly splashes banners. The first thing I noticed is that Netstat showed several connections to Foreign Addresses via port 80 TCP. One of the connections looked like the one HTTP-Tunnel Client was using to connect to its server on the Internet, 64.224.202.127:80 TCP. I did a Tracert to this address and found its DNS entry to be mail.rentmontreal.cc, not what I expected. I stopped the tunnel then started it again, same result. I wonder if they know they have been coded into this application.

To get Napster to use the HTTP-Tunnel Client you must configure Napster's Preferences. Within the Napster application select the proxy tab. Set the proxy type to 'SOCKS5', Proxy IP to '127.0.0.1', the port to '1080', and select 'Download files through proxy.' With the HTTP-Tunnel running I started a connection with Napster. Watching 'netstat –a –n 1' I saw a new series of port activity; Foreign Address of 64.224.202.115:80 TCP, a Local Address of 127.0.0.1:1080 TCP to a Foreign Address of 127.0.0.1:1766 TCP and another on 1771 TCP, a Local Address of 127.0.0.1:1771 TCP to a Foreign Address of 127.0.0.1:1080 TCP. Meanwhile the HTTP-Tunnel Client shows that it is establishing a connection with 64.124.41.171:8875 (Napster).

Connection Flowcharts:

Normal Napster, behind our Firewall;





The two arrows between nodes represent a two way TCP connection.
Our Firewall policies only allow Private Network Clients to initiate sessions with Internet (External) Servers, but never allow External Clients to connect to Private Network Servers.
Note: any given node can act as a Server or a Client, the relationship has more to do with who is requesting data.
When HTTP-Tunnel is running this policy is effectively circumvented.

Napster via the HTTP-Tunnel;

SOCKS5: SOCKS5 is an application offered by NEC Inc. marketed interestingly enough as a firewall.(5) The following is an excerpt from NEC's FAQ page. (Grammatical mistakes are on NEC's part, I am sure I have my own)

> "SOCKS is networking proxy protocol that enables hosts on one side of a SOCKS server to gain full access to hosts on the other side of the SOCKS server without requiring direct IP reachability. SOCKS redirects connection requests from hosts on opposite sides of a SOCKS server.

> The SOCKS server authenticates and authorizes the requests, establishes a proxy connection, and relays data."(NEC)(5)

I find myself asking the question, "Why is their firewall defeating my firewall." SOCKS, simply allows you to send all port traffic through one designated port. SOCKS, is similar to a VPN connection without encryption (planed for the next version). In our case SOCKS appears to be built into the HTTP-Tunnel, however HTTP-Tunnel may have its own proprietary code that acts very similar to SOCKS. Here is how it works, HTTP-Tunnel is installed and running, Napster is configured to use SOCKS5 as its proxy server with an IP address of 127.0.0.1 port 1080 TCP. HTTP-Tunnel listens to port 1080, picks up the packets sent to 127.0.0.1 then forwards them to the HTTP-Tunnel Server out on the Internet via port 80. Once the HTTP-Tunnel client/server relationship is established anyone within the Napster community will have file access to this computer. Theoretically that access is restricted to the folders the user has identified in the Napster preferences.

Conclusion: What this means is our current policy of blocking applications via their TCP/UDP port is beginning to look futile (that probably wont surprise anyone). We've had suspicions this was possible when we installed the firewall, but chose to ignore the problem until it became an issue. We dream someday of having the time and money to be proactive, but for now reactive is the norm. When Napster was still fairly new I was able to open a port other than Napster's default 8875, connect to a proxy server than connect to Napster. The port I opened on the firewall was something like 1750/TCP. I don't remember the specifics, but the point is that we knew tunneling through the Firewall was possible early on. At a minimum we decided never to open 1750. Immediately I suspected it would only be a matter of time before either Napster changed their software or someone would develop a proxy server that used port 80 to tunnel through. Unfortunately it happened, and the students have figured it out.

If you read between the lines there is a bigger problem than Napster. What if a student has setup a Napster/HTTP-Tunnel, and someone on the External network uses one of these as a backdoor to your Private network. How could they do this? Maybe disguise a back-orifice client as an MP3 file. Napster has a routine that only allows MP3/WMAs to pass through the community, so you would need to be creative. After the host MP3 is downloaded and played the executed back-orifice sends a short message to its server via port 80 or even the existing HTTP-Tunnel running on that computer. The back-orifice server responds and your private network is compromised. There may even be a way to use the HTTP-Tunnel directly to gain access to the Private network, perhaps an unscrupulous HTTP-Tunnel server operator. There is another Internet application GNUTELLA, that maybe a much greater risk. Sort of a Napster, HTTP-Tunnel, VPN solution all raped into one easy to use free application, but that's another paper.

Will we finally ban Napster at this University? Will we ban Tunnels? I think the tunnels are far more serious than the potential bandwidth problem associated with Napster. We are currently looking at setting up prioritization rules for traffic passing through our Cisco routers on both sides of our firewall. I strongly believe that to truly secure a network every aspect of that network must be under control. Computers would have a fixed load-set that could not be changed by the end user. The Internet would be accessible, however applications would not be downloadable. Only computers authorized on the network would be connected to the network. There are reasonable technologies and methods for implement this, however this does not fit the end users expectations, especially at a University. Our users would rather do what they care and have someone else consider the risks. As for TCP Tunnel servers using ports you can't block like port 80, find them, find them all, and block there IP addresses. Just remember to add this to your network security policy.

References

1. http://www.cis.ohio-state.edu/htbin/rfc/rfc1631.html, K. Egevang, P. Francis, May 1994

2. http://www.isi.edu/in-notes/rfc1597.txt, Y. Rekhter, B. Moskowitz, D. Karrenberg, G. de Groot, March 1994

3. http://www.napster.com, Napster, Copyright 2000 Napster Inc.

4. http://http-tunnel.com/newpage/icqp.htm, HTTP-Tunnel, Copyright I-SIGHT Design 2000.

5. http://www.socks.nec.com/socksfaq.html#q1, NEC, Copyright 1996-1999, NEC USA, Inc.

Last Updated: May 23rd, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event