Using IDS to Evaluate Outbound Port Usage for Security and Reduction of IDS Alerts A Case Study

Kenneth Underwood

Using IDS to Evaluate Outbound Port Usage for Security and Reduction of IDS Alerts, A Case Study.

Kenneth Underwood
GSEC Assignment 1.4, September 2002

Abstract
After recently deploying an Intrusion Detection System (IDS) inside our corporate LAN, the issue at hand quickly became apparent, reduction of the amount of alerts that appear to be part of normal traffic. Tuning the IDS or even the network itself to eliminate these alerts is the hardest part. I can see how an IDS Administrator might turn off certain categories of alerts, because they are so numerous that they become an annoyance. One such type are ICMP alerts. After all, in the entire scheme of things, ICMP might appear to fall short on the importance scale, when weighed against buffer overflows, attempted root access and other types of hacking exploits. With the reluctance to give in so easily, I tried to find out the cause of these alerts as many of IDS administrators will attempt to do as well. After a few hit and miss attempts, it started to become clear that some of these could be related to outbound port usage, and that the network border could be misconfigured. I decided to study the destination, or outbound, port usage of our internal workstations so I could be as informed as possible when addressing the alerts, and possibly the network border itself.

Finding opinions on the Internet about securing or blocking ports at a firewall or on other devises is not hard. Just doing a query on your favorite search engine will result in many examples to choose from. Invariably, the consensus that you will find again and again is that you should close all ports that you do not need. For the most part, the attention is drawn to inbound access to your internal network. Less can be found on outbound port blocking. Unfortunately, for the average Network Administrator that is new to hands on security, and might have to squeeze in some security along with many other duties they have, information about port blocking on the Internet can seem somewhat gray, and in enormous quantity. Taking advice about closing this outbound port, or that outbound port from someone you don't know, causes hesitation, or most likely, no action at all from the Administrator. With all the responsibility that the average Network Administrator has, "experimentation" at the network border is probably not on the job description. Using a flexible Intrusion Detection System can take the guesswork out of the equation. "Knowing" what traffic is leaving your network, is like turning on the light, where there was once darkness. This paper will give examples of what I found in our corporate network, and what I did about it.

The Setup

For this study, I used an IDS that is truly flexible. Snort[i]. This gave me the advantage of quickly and easily writing rules that triggered alerts as if they were hacking signatures or vulnerabilities. Some rules were applied for a matter of minutes, some for days, depending on the volume of alerts from the IDS. I will only focus outbound traffic, and how it might effect security, security policies or tuning the network to reduce IDS alerts. In the end, I hope this to be a high level view of ports and there effect on security and IDS alerts. If we do eventually end up blocking some outbound ports, perhaps we will learn why in the process.

I also want to mention that when I refer to "we" when it comes to configuring the Firewall and border router, I am referring to myself and our WAN group that has configuration responsibility. This also was a good lesson when dealing with other group, having documentation and supporting articles makes the process of conveying your message much easier. First, let's look at port numbers in general, then we will look at a few specific port usage scenarios.

Ports, In General

A continually updated database of port numbers can be found at the Internet Assigned Numbers Authority[ii], or IANA.

Ports have a range of 0-65535. They are divided into three ranges.

Well-Known ports 0-1023.

Registered Ports 1024-49151.

Dynamic/Private Ports 49152-65535.

If IANA's brief explanation falls short, just try a query on your favorite search engine.

Port usage from the workstation perspective.

When a user types abc.com into a web browser and hits enter, a simplistic view is that the browser needs to find the IP address of abc.com before it can communicate with abc.com, and eventually display the web page. This is done by querying a DNS server for the IP address. A workstation sends a query to the DNS server, destination port 53. But what is the DNS Servers' destination port when it sends it back? The answer is the original source port chosen by the workstation. The client source port used is greater than 1024. A good explanation of this is in the "Network Intrusion Detection, An Analysts Handbook"[iii] (If you are working with IDS, it will be invaluable to you.) Once the workstation knows the IP address of abc.com, it sends an HTTP GET to that IP address requesting a specific web page, most likely to destination port 80. But what of the abc.com web server? Assuming that is listening on port 80, it sends the page back to the computer that requested it, but to what port. Again, it's in the range found above 1024. So in the above scenario, can we block all outbound Well-Known ports (0-

1024) except 53 and 80 and have no detrimental effect on web surfing? Even knowing what I know so far, I am not willing to guess.

Special Applications
Does your company run specialized applications that make use of external computers to retrieve or transfer data over the Internet? They might be used to look up financial information, insurance history or driving records. Whether or not the data is encrypted or not is another issue, but you need to know its there first. I have discovered outbound traffic that the IT Staff was not aware of. This probably happens more then we think in small organizations where the IT staff is just trying to keep up with the users needs, and security does not fall under their responsibility.
Which destination ports do they use? Shall we just guess?

Port usage from the Network Services perspective.
Of course, mail gets delivered to your mail servers port 25, that is why port 25 is open inbound to your mail server. During that inbound port 25 connection, what port is it using outbound to the sender? When sending out mail, does the destination mail server communicate back on the same port 25?
Your web server listens on port 80, but on what port does it send back out?
What if you start blocking outbound ports, what are the implications?
Start blocking outbound ports without being informed, and embarrassment will be the least of your worries.

Port usage from the protocol perspective.
We have talked about well known services like DNS, Web and Mail services and what ports they are associated with. Does your network use any others, like Netbios, RPC's or any others? If it does, where does it go?

Network Infrastructure and the placement of the IDS.
In this study, only one point of access to the Internet was available to 35 internal subnets. To capture traffic to and from the Internet, the IDS is attached to a port on the same switch that a Firewall and internal router is also attached to. Any traffic that was destined to or from the Firewall was in essence, copied and sent to the IDS as well. This is known as port spanning. At the Firewall, Network Address Translation was applied, changing the internal private IP addresses to an Internet routable IP address range. The data that you will see is before that translation takes place, and the address translation is irrelevant to the data, and would only cloud the issue. The Firewall only allows unsolicited inbound traffic to one Server, that is the Mail Server and port 25. In effect then, all inbound ports are blocked except port 25. No Web Server is hosted at this site.

The Method
If your IDS is as flexible as Snort, you should be able to define a rule to capture outbound packets based on port usage. Snort provides a very easy way to generate and revise a rule that will trigger an alert for almost anything you can think up. Lets break down a typical rule that I used and identify the pieces.

alert ip $HOME_NET any -> $EXTERNAL_NET 81:109
msg:"Port Usage" sid:1000001 rev:27

Alert: This tells Snort what to do once it detects the traffic you have asked to see. There several actions besides "Alert" that Snort can take, including just logging the event. So if you don't want all of the alerts clogging console, you can just log them, and the query the database later to gather your data.

IP: This tells Snort what protocol to look for. Since I wanted to see both TCP and UDP packets, telling snort to look for IP packets will capture both.

$HOME_NET: I have already identified my internal network IP scheme to Snort in the variable section. In our example, it would be 172.16.0.0/16. We use an internal class C scheme, but I tell Snort to mask it like a Class B. Using this method, the third octet in the network portion can be any subnet we wish, and we can add new subnets without effecting the $HOME_NET variable.

Any: This tells Snort look for a source port of "any" range, meaning all of them. I am not really looking at source ports and I don't want to restrict this to a specific range, or Snort will ignore whatever is not in that range. After all, the source port is only really relevant here when the packet returns to our network from the Internet.

->: This is the direction operator, and tells Snort to only generate an alert if the direction is outbound. I do not want to include the inbound traffic, for it brings into play the source port again.

$EXTERNAL_NET: Also in the variable section, I have told Snort that the external network, or the Internet, is anything that is not the $HOME_NET. Multicasts and Broadcasts show up since I have not identified 224.0.0.0 or 255.255.255.255 in our $HOME_NET variable.

81:109: In this revision of the rule, I have asked Snort to generate an alert if the destination port is between 81 and 109. When I first wrote this rule, I had made the range 81 though 442, expecting to see no traffic between normal web traffic on port 80 and SSL on port 443. I was quickly wrong, as I saw POP traffic, Time Services and some others almost immediately. So I backed it down to 109 so I could run it for a few days and try to be fairly sure that this port range was not used on a regular basis. Only after that, did I revise the rule to look for only "110"

for a few days to see who was using POP, then changed the rule again to a range of 111:122. Knowing that I saw the Time Service using port 123, I could then run this new rule for a few days to make sure those ports were not used on a regular basic as well, before revising the rule again to capture port 123 traffic. I continued doing this again and again in small pieces, until I reached the entire port range of 65535.

msg:"Port Usage": This is the alert message that would be displayed on the console. It could say anything you wish. I could have identified the port range I was using at the time, but I already knew what range I was looking for during each phase, and did not bother to change it for each revision.

sid:1000001 rev:27: This just refers to the "Snort ID" for the rule, and its revision number that changed each time I changed it. I could use the same rule over and over and just change the port range that I was working on.

Here is an example of a packet captured using destination port 80
Event Data
Message Port Usage Generator ID 1
Classification 0 Snort ID 1000001
Priority 0 Revision 26
Rule Active alert ip $HOME_NET any -> $EXTERNAL_NET 80 (msg:"Port Usage"; sid:1000001; rev:26;)

IP Header
Version: 4 Header Len: 5 TOS 0 Total Len (in bytes) 606
16-bit ID 17724 Frag Flags DF 13-bit offset 64
TTL 126 Protocol TCP
Source IP 172.16.x.x
Destination IP 64.58.76.179

TCP Header
Source port: 1795 Dest Port: 80
Sequence Number 0x19E3C4F
Acknowledgement Number 0xB938141A
Offset 5 Flags ***AP*** Window 8760 Urgent Pointer 0

Packet Payload
Payload Length: 566 bytes
(Not Shown)

There is more data in the payload, but I think you can see the type of data it can display.
There are various techniques that you can use to investigate the identity of a destination. A good start would be a "whois" type lookup. All IP address assignments can be found in one of 3 Internet Registry sources.

ARIN[iv] (Americas and Sub-Sahara Africa)
APNIC[v] (Asia/Pacific Region)
RIPE NCC[vi] (Europe and Surrounding Regions)

Starting at ARIN, it was easy to identify this destination as a Yahoo search page, and quickly categorize this traffic as normal. Other destinations might be harder to identify, such as Peer-to-Peer file sharing, where the destination is a workstation with no public service like a web site, and the reason for the session is not obvious. You can return to the IANA site to try and find the reason for the traffic based on the port number itself. If the IP address is registered to AOL for instance, with the addition of the port number, you might surmise that is it an AOL client generating the traffic. If you identify the destination, but the port number seems odd, you could write a rule that captures all traffic from a specific internal IP address, to the "$EXTERNAL_NET" and hope its not too much data that it makes it too much to look through. Of course, you could work directly with the internal employee to try and find out if the traffic is generated only when they use a specific software program. Also, Google[vii] is a great tool for investigating traffic, especially the groups section. If you are new to this type of investigation, a good place to start is a paper named " Techniques for Identifying the Threat to your Systems from Researching the Apparent Source of an Attack"[viii].

That should give you an idea of the method I used to gather the data.
A complete explanation of the Snort rules language can be found in the Snort Users Manual[ix].

The Data
Briefly, an explanation of the table layout:
Dest. Ports = Destination port or port range at the TCP layer.
Service = The expected service associated with a specific port. The term "Many" is used only when no traffic was seen and listing all services in that range would make the table to large.
Protocol = The protocol most likely seen using this port.
Packet Count = The number of packets captured during the rules activation time. The time it was active was just enough time to get a good amount of data, and the count itself is less important then the suggestion that the port is used widely or sparingly.
Block? = In final analysis, would we block this outbound port.
Source = The IP address of the internal workstation.
Computer Name = Included to give the analyst an easier way to ID workstations.
Destination = The sanitized IP address of the target.
Internet Domain = The sometimes sanitized domain name of the target.
Reason = Why the source is communicating with the target.

Dest. Ports	Service	Protocol	Packet Count	Block?	Source	Computer Name	Destination	Internet Domain	Reason
0		ICMP	4	no	Techs	Techs	many	Many	ping
1-19	many		0	no	0	0	0		
22	SCP	TCP	10000	no	172.16.199.198	Techs	x.x.107.170	Trusted Partner	File Transfer error_log
23	Telnet	TCP	641	no	172.16.69.14	Linda5	x.x.161.2	Trusted Partner	File Transfer
24	Private Mail System		0	no	0	0	0		
25	smtp	TCP	141	no	172.16.199.242	Mail Server	many	Many	Mail Transfer
26-41	many		0	no	0	0	0		
42	Host Name Server	TCP/UDP	36	no	172.16.199.205	NT5	224.0.1.0	None	Wins Discovery
42	Host Name Server	TCP/UDP	36	no	172.16.199.222	NT1	224.0.1.0	None	Wins Discovery
43-52	many		0	no	0	0	0		
53	DNS	UDP	556	no	many	many	x.x.121.66	DNS1	Dns Lookup
53	DNS	UDP	45	no	many	many	x.x.186.2	DNS2	Dns Lookup
54-66	many		0	no	0	0	0		
67	BootP Server	UDP	2	no	172.16.199.63	Joe	255.255.255.255		Find BootP Server
68-79	many		0	no	0	0	0		
80	http	TCP	4505	no	many	many	many	Many	Web Traffic
81-109	many		0	no	0	0	0		
110	Pop	TCP	389	no	172.16.199.25	Fred	x.x.0.101	POP ISP1	Pop Mail access
110	Pop	TCP	254	no	172.16.199.26	Joe	x.x.0.100	POP ISP2	Pop Mail access
110	Pop	TCP	268	no	172.16.70.60	Dave	x.x.121.99	POP ISP3	Pop Mail access
111-122	many		0	no	0	0	0		
123	Time	UDP	24	no	172.16.199.222	NT1	x.x.198.40	1.NAVY.MIL	Atomic clock
123	Time	UDP	24	no	172.16.199.24	Jack	x.x.41.209	2.navy.mil	Atomic clock
124-133	many		0	no	0	0	0		
134	ingres-net		0	no	0	0	0		
135	RPC/DCE Endpoint resolution	TCP/UDP	124	yes	many	many	Mis-Config		Bad IP address
136	PROFILE Naming System		0	yes	0	0	0		
137	NETBIOS Name Service	TCP/UDP	434	yes	many	many	many	Many	
138	NETBIOS Datagram Service	UDP	166	yes	many	many	169.254.172.167	MS AutoPriv IP Addr	Cant Find DHCP
138	NETBIOS Datagram Service	UDP	166	yes	many	many	x.x.121.66	DNS1	Netlogon And Browsing

139	NETBIOS Session Service	TCP	18	yes	All NT	All NT	x.x.121.66	DNS2	File/Print Sharing
161	SNMP	UDP	8	yes	172.16.199.27	Jet Direct	255.255.255.255	Broadcast	SNMP Discovery
162-199	many		0	no					
200-426	many		0	no	0	0	0		
427	SLP		15	no	172.16.199.63	Joe	224.0.1.22	Multicast	SLP Registration (SAP for IP)
426-442	many		0	no	0	0	0		
443	SSL	TCP	122	no	many	many	many	Internet	Encrypted Http
444-519	many		0	no	0	0	0		
520	RIP	UDP	414	no	172.16.199.99	Router1	255.255.255.255	Intranet	RIP
520	RIP	UDP	1139	no	172.16.199.96	Router2	255.255.255.255	Intranet	RIP
521-1023	many		0	no	0	0	0		

The Analysis and Action

Let's work our way down the previous chart.

The IT Staff needs to ping various targets, so blocking ICMP outbound, will only take a tool away from the IT Staff. A port is not used here, but a rule was easily written to capture this traffic, so it was included.

No traffic was seen using ports 0 through 21.

Port 22 is used for a Secure Copy over an SSH connection, this will be needed by the IT Staff as well.

A Telnet (port23) session from an end user was surprising. This could be a security risk if any confidential information is transmitted or received over the Internet. After contacting the software Vendor, a more security minded application (HTTPS) had already been developed and the user started using it immediately. Although we don't want to block this port either, due to possible IT Staff usage, it did bring to light an issue that we were not aware of.

No traffic was seen using port 24.

For Port 25(SMTP), this is expected. We certainly do not want to block this destination port. Only traffic from the mail server was seen using this port. This IDS rule only ran for about 5 minutes.

No traffic was seen on ports 26 through 41.

Port 42 was the destination port for a Wins discovery multicast. This was news to us. But the Router will not allow this to leave that subnet, and the traffic count was low, so we will do nothing about this at this time.

No traffic was seen on ports 43 through 52.

DNS uses destination port 53, considering that our ISP hosts this service, blocking this would disrupt surfing the Internet.

No traffic was seen using destination ports 54 through 66.

We found one machine that was looking for a BootP server at boot up, this was not destined for the Internet, and was a broadcast that the IDS saw only because it was on the same subnet. I advised the user, which was a member of the IT Staff, and it stopped.

No traffic was seen using destination ports 68 through 79.
Destination port 80 can not be blocked either, or surfing would end.
No traffic was seen destined for ports 81 through 109.
Port 110 (PopMail), was seen from a few workstations. Primarily due to pulling in POP home accounts into the Outlook program. We are somewhat confidant that due to a layered virus defense, specifically virus protection at the Mail Server, workstations and at the Servers, that proliferation from a virus would be stopped. Malicious code on the other hand, could be a problem, and a warrants further study.
No traffic was seen using destination ports 111 through 122
A few NT Servers that were synchronizing with an atomic clock were using Port 123 (Time) service. Internal workstations would then sync with the NT Servers. So this is a needed service.
No traffic was seen using destination ports 124 through 133
Port 134 (ingres-net) saw no traffic.
Port 135 (RPC/DCE endpoint resolution) saw 124 packets in a short period of time, but not in the way I found the others. In an effort to confirm my data, I placed Snort between the Firewall and border router for a few hours, testing various ports. Although when I did the original study, I found that no traffic was destined for the $EXTERNAL_NET using this port, or was it. Lets briefly revisit how I defined the $EXTERNAL_NET to begin with. I told Snort that the $EXTERNAL_NET was what "was not" the $HOME_NET. I Defined the $HOME_NET as 172.16.0.0/16 so that all our subnets defined in the third octet would be included. The issue arose when the Data Center, which has its own IT Staff, deployed a Web Server with a normal IP address for our network, but that Web Server queried a Oracle database that was connect only to the Web Server, eliminating the need for all of the clients to need a connection license. This database server was deployed with a subnet undefined in our routers. Somewhere in the client software, that database IP address was referenced, and the workstations tried and failed to RPC to the database server. This did not effect the operation of the software at all, because the Web Server ultimately knew how to find it since it was connected directly to it. The RPC on the other hand was sent to the routers default gateway, the Firewall. The Firewall continued sending it past the border router and out to the Internet. Remember that one of the goals of this study was to reduce ICMP alerts. This was the cause of over 1000 per day back from the Internet. ICMP code 13 (Destination Unreachable, Administratively Prohibited) was returned on a regular basis from our ISP, which of course would not route a private address scheme on the Internet. So why did a private IP address reach our ISP? This was due to the border router not having a comprehensive egress filtering rule applied. On Sans' "The Twenty Most Critical Internet Security Vulnerabilities"[x], number G5 (Not filtering packets for correct incoming and outgoing addresses) states among other things, "Any packet leaving your network must not have a destination address of your internal network"[xi]. Also, when a member of the IT Staff found a program that scanned for SNMP ports listening on our entire private IP address range (172.0.0.1 through 172.255.255.255) any subnet that was not specifically

entered into the routing table for the internal router, was sent to the routers default gateway. It made it to the first hop outside our network, our ISP, and the ISP router rejected it, sending back an ICMP unreachable. You can imagine how many of those we got that day.
This prompted a full evaluation of the border router configuration. Egress filtering and other security related items beyond this study, were applied, reducing over 1000 ICMP alerts per day on the IDS. We also blocked port 135 outbound on the Firewall for good measure, as RPC's to the Internet are not required as proven in the first round of our port study.
Port 136 (Profile Naming System) saw no traffic.
Ports 137 saw a lot of traffic in a short period of time. This was also another major cause of ICMP back from the Internet in the form of ICMP type 3, code 3 (Destination Unreachable, Port Unreachable). This stemmed from bad HTML code on a popular web page. Within that web page was a stock ticker that fed a stream of stock data to the browser. For whatever reason, all of the workstations did a Netbios lookup to this Ticker Server on the Internet. Once the lookup reached the Ticker Server on port 137, no service was listening on that port, and it returned the Port Unreachable to the internal clients. This caused about 400 of these alerts to trip the ICMP rules on the IDS per day. So do we really need netbios lookups over the Internet? Logic says no, so we blocked this outbound port at the Firewall as well.

Port 138 (Netbios Datagram Service) saw some traffic as well. One of the destination addresses was 169.254.172.167. This was not to the $EXTERNAL_NET, but since it was not defined as a $HOME_NET variable, and was not defined in the internal routers routing table, it was also sent to the routers default gateway and out to the Internet as well. This caused ICMP Destination Unreachable alerts at the IDS as well, one for each occurrence.
So what is 169.254.172.167? This is the automatic IP addressing scheme for Windows 2000 machines that can not find the DHCP server at boot up time. This was due to the Data Center project as well, and was corrected early on. But again this illustrated the lack of a comprehensive egress filtering process at the border. I worked with the WAN group to stop this traffic from leaving our network, but also blocked port 138 at the Firewall for if no other reason, port 138 is not needed on the Internet either.
Port 139 (Netbios Session Service) was about the same amount of traffic as port 138, plus the NT boxes were seen sending this to the DNS servers for File and Print Sharing activity. I have not absolutely determined the reason for this, but logic tells me that this is definitely not needed. Port 139 was blocked at the Firewall as well.
Ports 140 through 160 saw no traffic
Port 161 (SNMP) saw minimal traffic to a broadcast address. This turned out to be a Jet Direct card scanning its neighborhood for other SNMP devices. This was a default installation, and the issue is well known as I found out searching the Internet for this behavior. It was easily stopped in the Jet Admin software. But we have no need for this, and will block this outbound port. If any internal machine

were compromised, scanning others on the Internet from our LAN, to this destination port, would be denied.

Ports 162 through 426 saw no traffic.
Port 427 was seen again on an IT Staff computer. This turned out to be related to a Novell version of Client for NetWare, that was broadcasting a SAP advertisement. Again, since this was not destined for the firewall, and only seen because the IDS was on the same subnet, nothing was done about this.
No traffic was seen using ports 428 through 442.
Port 443 (SSL) traffic was seen, and secure surfing would be disabled if this port were blocked.
No traffic was seen destined for ports 444 through 519.
Two internal routers were seen broadcasting to port 520, or RIP. This is normal behavior.
Ports 521 through 1023 saw no traffic.

Now in regards to the port usage so far, we could really make a short list of all of the needed ports. The ports used can be explained by associating them with a specific service, or putting them in the anomalous behavior category.
For a look at what is closer to the unexplained, lets look at the rest of the data.

The Rest of the Data

Dest. Ports	Service	Protocol	Packet Count	Block?	Source	Computer Name	Destination	Internet Domain	Reason
1080	Socks	TCP	8	no	172.16.69.15	Dan	x.x.160.32	Research Site	
1214	Kazaa	TCP	548	yes	172.16.199.63	Joe2	many	Grokster	File Sharing
1214	Kazaa	TCP	164	yes	172.16.199.61	Tom	many	Grokster	File Sharing
1214	Kazaa	TCP	90	yes	172.16.199.25	Fred	many	Grokster	File Sharing
1414		TCP	2	no	10.10.70.16	Don	x.x.28.245	Trusted Partner	
1414		TCP	2	no	172.16.198.132	Ellem	x.x.28.245	Trusted Partner	
1414		TCP	2	no	172.16.198.13	Donna2	x.x.28.245	Trusted Partner	And others
1755	Streaming	UDP	8	no	172.16.61.55	May	x.x.219.7	Broadcast .com	Internet Radio
1863		TCP	4	no	172.16.70.13	Rick	x.x.12.134	Hotmail .com	Hotmail
1892		TCP	3	no	172.16.198.2	Barb	x.x.212.253	ftp.nai.com	
1892		TCP	2	no	172.16.199.242	Mail Server	x.x.95.24	Mail Server	Outbound email
2080		TCP	28	no	172.16.69.12	Ann	x.x.215.50	Trusted Partner	
2112		TCP	4	no	172.16.68.1	Brian	x.x.227.239	ca.sportsline.com	
3443		TCP	9	no	172.16.69.12	Ann	x.x.215.50	Trusted Partner	
3993		TCP	2	no	172.16.9.2	SSR2	x.x.225.4	treas.gov	
3999		TCP	1	no	172.16.70.23	Rick2	x.x.126.156	ISP	
4002		TCP	1	no	172.16.70.23	Rick2	x.x.126.156	ISP	
5050		TCP	4	no	172.16.199.49	Comp	x.x.233.128	yahoo.com	

						Room			
5190		TCP	13	no	172.16.199.49	Comp Room	x.x.69.137	dial. aol.com	AIM
5282		TCP	4	no	172.16.199.1	Danny	x.x.102.89	Trusted Partner	
6346		UDP	335	yes	172.16.199.24	Jack	many	many	GNUTella FileSharing
6502		TCP	8	no	172.16.199.114	850_1	255.255.255.255	Broadcast	
6502		TCP	8	no	172.16.199.119	WS-8	255.255.255.255	Broadcast	
6502		TCP	8	no	172.16.199.112	850_2	255.255.255.255	Broadcast	
7023		TCP	4	no	172.16.60.105	Ray	x.x.182.100	Trusted Partner	Insurance Lookup
8000		TCP	15	no	172.16.199.23	Rick1	x.x.88.10	ngi.it	
8080		TCP	19	no	172.16.60.33	George	x.x.126.212	Temp Domain name.com	Yahoo Messenger
8080		TCP	4	no	172.16.70.64	AMY	x.x.33.220	ads.com	
8080		TCP	2	no	172.16.69.20	Pat	x.x.124.40	site.net	Yahoo Messenger
8080		TCP	4	no	172.16.70.23	Rick2	x.x.157.101	games-world.net	
8111		TCP	6	no	172.16.70.66	Deb	x.x.238.73	Trusted Partner	
8194		TCP	5	no	172.16.67.31	Kit	x.x.212.143	Trusted Partner	
8194		TCP	2	no	172.16.67.31	Kit	x.x.53.143	Trusted Partner	
8292		TCP	7	no	172.16.69.25	Org	x.x.53.157	Trusted Partner	
8292		TCP	2	no	172.16.67.31	Kit	x.x.53.157	Trusted Partner	
8294		TCP	4	no	172.16.67.31	Kit	x.x.250.45	Trusted Partner	
8383		TCP	102	no	172.16.68.26	James	x.x.0.101	ISP	Web mail
8900		TCP	105	no	172.16.18.2	18Mgr	x.x.233.41	School .edu	Distance Learning
8999		TCP	10	no	172.16.60.105	Ray	x.x.182.100	Trusted Partner	Insurance Lookup
9536		TCP	440	no	many	Many	x.x.138.72	Trusted Partner	Vendor Order
32328		TCP	4	no	172.16.68.43	Tim	x.x.24.141	America Online	AIM?
32328		TCP	1	no	172.16.68.43	Tim	x.x.161.185	blue. aol.com	AIM?
32328		TCP	1	no	172.16.68.43	Tim	x.x.26.38	America Online	AIM?
32328		TCP	1	no	172.16.68.43	Tim	x.x.161.153	blue. aol.com	AIM?
41178		TCP	4	no	172.16.199.63	Joe2	x.x.58.81	Audio Galaxy .com	
48129		TCP	5	no	172.16.69.25	Org	x.x.53.131	Trusted Partner	
48129		TCP	1	no	172.16.67.31	Kit	x.x.53.131	Trusted Partner	
many		TCP	many	no	many	Many	x.x.2.7	nai.com	ftp get dat files
many		TCP	many	no	many	Many	x.x.212.253	nai.com	ftp get dat files
many		TCP	many	no	many	Many	x.x.28.148	nai.uk	ftp get dat files

You might expect to see little traffic on destination ports 1024 and up. As you can see quite a bit appears. Peer-To-Peer file sharing, custom applications, web surfing to non-default ports, Instant Messenger, Ad Servers, WebMail, gaming, steaming audio, remote software discovery broadcasts and ftp were some of the reasons that we saw traffic above port 1024.
With the mix of ports so wide, practicality comes into play. With 64513 ports above 1023, blocking groups of ranges could become a maintenance problem. Doing what is "reasonable" is the required action here, although you might take a more restrictive response. Many of these ports being used are critical to the employees. They should be evaluated on a case by case basis.

The security policy also comes into play here. Although the security policy uses the "catchall" phrase that "only the It Staff may install approved software", what if the IT Staff are the ones using some of these programs to begin with. As far as file sharing software and Instant Messaging goes, the approach I chose was to make available to the IT Staff a number of articles that outlined the risks. I gave them an X-Force Whitepaper called "Risk Exposure Through Instant Messaging And Peer-To-Peer (P2P) Networks"[xii], and information about how a Gnutella can be a pawn in a Firewall subversion technique that will force the internal workstation to initiate a separate outbound connection to a hacker.
I was unable to find a document that was publicly available on the Internet, so I will not outline this vulnerability.
Before reading these documents, half of the IT Staff was willing to say that as long as you know what you're doing and know what files your sharing, that it was safe. They said that of course they would not trust the average user to know how to protect themselves and the network, but the average user is not a highly trained technical machine, like us.
I joke of course, but really their sentiment was not far off. I can kid them, as I am a member of the IT Staff. But after reading the documents mentioned above, not one was willing to go on record as being "for" file sharing in the corporate LAN. Ports 1214, for the connection to the Kazaa network, as well as 6344, 6345 and 6346 for the Gnutella network have been blocked. An informed user could change the default ports, so a Snort rule to detect these ports was activated for permanent use to detect a first time initial connection try before it reached the firewall and the blocked port. Any detection would be followed by a visit to the workstation to uninstall the software and inform the user.

Port 1892 looks to be the Mail Server. I think we can safely say that the Mail Server acts as any workstation during inbound mail transfer. A Mail Server somewhere in the world contacts our Mail Server on port 25. Then our Mail Server chooses a random port from 1024 or above to be its source port number. Acknowledgment are sent back to the sending Mail Server to its origin source port which it also chose randomly before it contacted our Mail Server, in this case 1892. If you hosted a Web Server, the traffic would be similar to this inbound mail delivery. Someone puts your URL into the browser and hits enter. They choose a random port from 1024 or up and send the HTTP GET to your port 80. Your Web

server chooses its own port from 1024 or up and send the page back to that someone, destined for that original source port.

The ftp on port 60001 also stood out. The destination was NAI.com. The workstations are configured to ftp updated Dat files from McAfee on a daily basis. So why port 60001? This, I believe, is due "Passive FTP" traffic. After the initial connection to the NAI ftp site on the default port 21, the server sends a PASV command to the client, the client then initiates an outbound connection to that new port. A good article on Passive Ftp is "Active FTP vs. Passive FTP, a Definitive Explanation"[xiii].
Port 60001 is also a port used by the Trinity Trojan or Distributed DoS tool.
I confirmed that NAI's ftp program uses the passive mode and the target of this port was always NAI's ftp site and that the packet payload carried the telltale sign "DELTA.INI" in its initial FTP GET. So I am confident that we do not have a Trinity tool scanning the Internet from within our LAN.

Other Ports

Another reason for blocking some outbound ports is that if one of our internal computers was ever compromised, and a Trojan installed, attacking or scanning computers on the Internet will be stopped at the firewall. As you can see with the last example with port 60001, we can not block all Trojan ports, as the list is very long and growing. It includes well-Known ports as well. A list of the common Trojan ports can be found at Sans[xiv].
The best we can do, if anything, is to keep an eye on the top ports being scanned over the Internet, and block any ports that have an unusually high amount of traffic on them.
A good list is at Dshield.org[xv] or at Incidents.org[xvi].
A few months ago, some Trojan ports were being heavily scanned, like SubSeven, now SQL has crept up and made the list. With that in mind, even though we saw no traffic to some destination ports, I added a few to the block list. For every port blocked, I added a permanent Snort rule that checks for the attempt to use the outbound port. If I see a hit on this, I can check to see if the user has a new software program that fails, or check the workstation for that Trojan.
These extra ports include:
LDAP, Port 445. TCP/UDP. This replaces the Netbios lookups for Windows 2000 machines.
SubSeven and others, Port 27374 TCP/UDP
SQL Query, Port 1433 TCP/UDP
Ring Zero, Port 3128 TCP/UDP
NETBUS, Ports 12345-12346 and 20123-20124 TCP/UDP
BACK ORIFICE, Ports 31337 and 54320-54321 TCP/UDP
You can learn more about each of these, just search on the Internet.

Summary of Action

- We found insecure software at a workstation and a more secure method was employed. One workstation was found to be misconfigured.
- We blocked service ports 135, 137, 138, 169 and 161.
- We have decided that we should investigate if malicious code vulnerabilities exist if POP mail is pulled into Outlook clients.
- We blocked file sharing default ports of Kazaa and Gnutella and made permanent rules to detect for the default port numbers, while turning around perceptions of some IT Staff to our side.
- Increased the knowledge of the IT Staff of what was leaving the network.
- Finding the border router loosely configured, we applied egress filtering at the border router to restrict private IP addresses from leaving the network. This eliminated thousands of ICMP packets back from the Internet that tripped Snorts' ICMP rule set.
- Forced to look closer at the configuration of the border router, we applied other security settings unrelated to this study, such as denying source routing, changing the SNMP default community string, "Public", disabling the Finger service, and applying an NTP vulnerability work-around.
- We blocked the default ports of a half-dozen popular Trojans just in case we are compromised, we are not a menace to others.

All this from the information gathered by detecting outbound port ranges.

Conclusion

With this knowledge about your own network now in hand, how much more informed will you be when addressing network traffic on your LAN? You can more confidently work with the WAN Administrator to close ports and apply egress filtering at the border, or deal with the IT Staff when it comes to security issues. Does your security policy need updating when it comes to Peer-To-Peer file sharing or Instant messaging and similar software?
And you will be able to reduce IDS alerts in the process.
Who knows what you might find. Happy Hunting!

---

[i] Snort Commercial IDS Sensor v1.8.3 & v2.0.
URL: http://www.sourcefire.com

[ii] The Internet Assigned Numbers Authority, IANA. PORT NUMBERS. 2002-08-28
URL: http://www.iana.org/assignments/port-numbers

[iii] Northcutt, Stephen; Novak, Judy. Network Intrusion Detection, An Analyst's Handbook. Indianapolis: New Riders 2000. 27

[iv] ARIN (American Registry for Internet Numbers) - Americas and Sub-Sahara Africa. 2002-09-02 URL: http://www.arin.net

[v] APNIC (Asia Pacific Network Information Centre) - Asia/Pacific Region. 2002-09-02
URL: http://www.apnic.net/

[vi] RIPE NCC (Réseaux IP Européens) - Europe and Surrounding Regions. 2002-09-02
URL: http://www.ripe.net/

[vii] Google. Search Engine. 2002-09-02
URL: http://www.google.com/

[viii] Lemmon, David. " Techniques for Identifying the Threat to your Systems from Researching the Apparent Source of an Attack" 2002-07-09
URL: http://rr.sans.org/incident/techniques.php

[ix] Roesch, Martin. "Snort Users Manual", Snort Release: 1.9.x 2002-09-02
URL: http://www.snort.org/docs/writing_rules

[x] " The Twenty Most Critical Internet Security Vulnerabilities (Updated), The Experts' Consensus" Version 2.504 2002-05-02 URL: http://www.sans.org/top20.htm

[xi] " The Twenty Most Critical Internet Security Vulnerabilities (Updated), The Experts' Consensus" Version 2.504 2002-05-02, Section: G5 – Not filtering packets for correct incoming and outgoing addresses URL: http://www.sans.org/top20.htm

[xii] X-Force Whitepaper. "Risk Exposure Through Instant Messaging And Peer-To-Peer (P2P) Networks" 2002-04
URL: http://documents.iss.net/whitepapers/X-Force_P2P.pdf

[xiii] "Active FTP vs. Passive FTP, a Definitive Explanation" 2002-09-02
URL: http://slacksite.com/other/ftp.html

[xiv] von Braun, Joakim. "Intrusion Detection FAQ" 2002-09-02
URL: http://www.sans.org/newlook/resources/IDFAQ/oddports.htm

[xv] Distributed Intrusion Detection System. 2002-09-02
URL: http://www.dshield.org/

[xvi] Internet Storm Center. "Top 10 Ports" 2002-09-02
URL: http://isc.incidents.org/top10.html