Global Information Assurance Certification Paper

Practical for GCIA Certification

GIAC Certified Intrusion Analyst

Written by
Andy Abercrombie

SANS 2001 San Diego
Assignment Version 3.0

Table of Contents

Assignment 1 – Describe the State of Intrusion Detection

StealthWatch by Lancope, Inc.

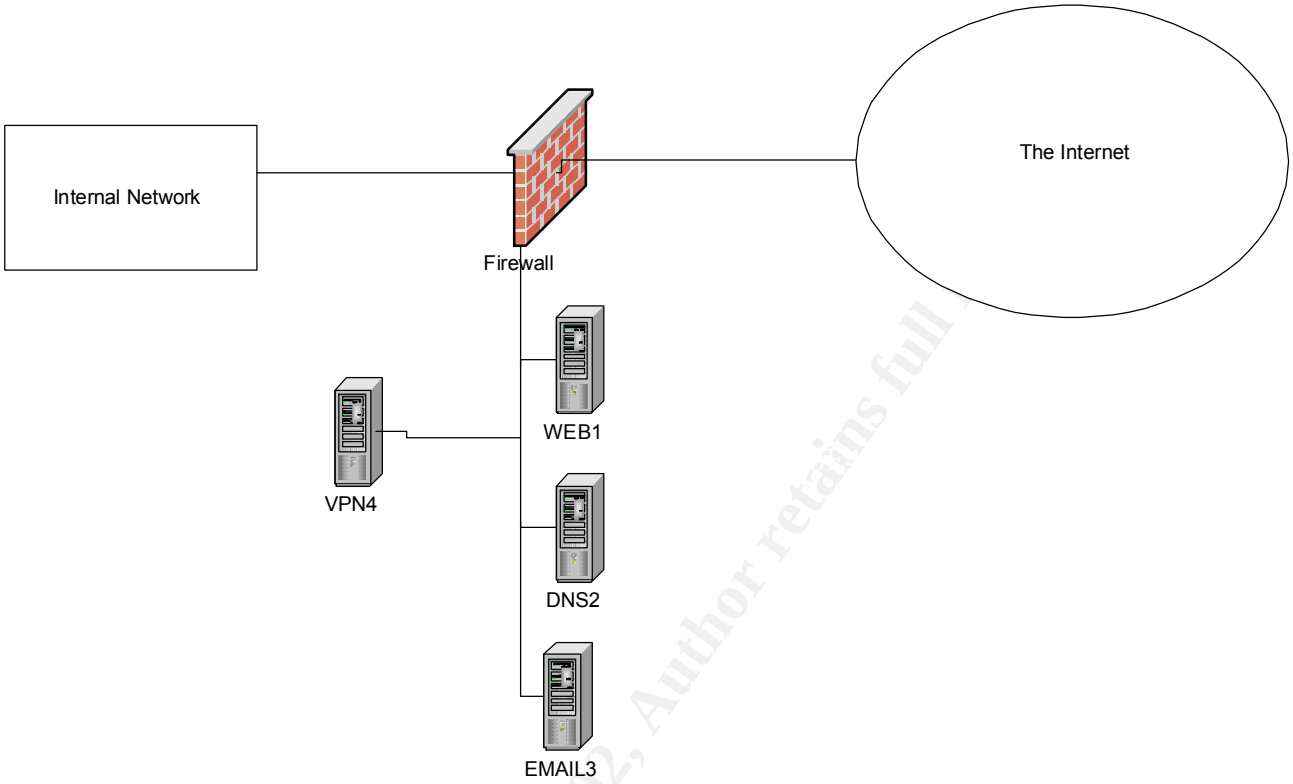
How do you detect intrusions that don't match a signature?

Today, most ID (Intrusion Detection) Systems use a set if signatures to compare against for abnormal traffic. If the traffic matches one of the signatures, then it has successfully detected an intrusion or matched suspected traffic to a known pattern. Now what if it doesn't match? Is the remaining traffic valid then? If you don't have a signature for bad traffic, how do you know it is bad? What if the malicious traffic can subvert a signature?

The increasing demand for a complete IDS solution has led companies to work on developing different ways to detect anomalous traffic. By base-lining your network segments, you can get a picture for how normal traffic behaves. You can begin to see what traffic is normal, therefore creating a profile to compare against in the future. In this manner, another method of ID is born. Looking at traffic data that is not normal, whether or not is matches a signature, can help you discover malicious traffic that was unknown, high-bandwidth traffic that was unknown, or damaged hardware. The advantages of this can be used beyond ID, but they are deeply root in ID as well. The appliance StealthWatch by Lancope is one such tool to do this.

What is a normal profile? If you take a look at your Internet pipe or network segment, you will see all kinds of traffic. Some of this is your normal corporate traffic (web surfing, ftp connections, or email relaying), some of it is management traffic (snmp, ssh, or VPN), some of it is protocol or hardware traffic (spanning tree, VRRP, or broadcast), and then there is the unknown. Unknown traffic can be bad or good. It just means it is traffic that is there and you don't know why. Malicious traffic would fall into this category along with misconfigured hardware (routers or switches) or protocols that are roaming around unknown (NetBIOS, IPX, AppleTalk). This is where the work begins.

If I took a look at a segment, I should be able to qualify the traffic according to the Corporate Network Policy for Allowed Use. If I take a copy of the policy, I can determine what protocols and transactions I should see on the wire. This will come in handy after I start sniffing the traffic and comparing it to "what is normal". Internal segments will be harder due to more bandwidth and less restrictions. It is easier to see this modeled in your Internet environment where protocols and allowed traffic are highly scrutinized and have to abide to strict corporate security policies.



On an Internet link, your routers and firewalls are configured to allow only valid traffic and to prevent access to unnecessary ports or machines. Using the picture above, here is an example of a simple service network setup off of a firewall. In this example, the company is offering a web server to the Internet, providing a DNS server for host name resolution, relaying email with an email server, and allowing clients to VPN into the corporate network. Knowing the firewall policy, we can determine that WEB1 should receive TCP port 80 traffic only from the Internet, DNS2 should be sending UDP port 53 to and from the Internet, EMAIL3 should be sending TCP port 25 to and from the Internet, and VPN4 should be sending and receiving IPSec (IP 50 and 51, and UDP port 500) traffic. This is a simple model for demonstration purposes, but you should be able to see the validation of anomalous detection soon. We now take our appliance for heuristic detection and place it outside the firewall. It is now going to monitor ALL traffic to and from the Internet. Also, since I have just laid out what ports and protocols are being used, we set the profile to note that the following traffic is normal.

Internet → WEB1	TCP port 80
Internet ←→ DNS2	UDP port 53
Internet ←→ EMAIL3	TCP port 25
Internet ←→ VPN4	IP 50,51 & UDP port 500

This is now the normal profile for what traffic we are going to see on this Internet link. This is how it was designed and configured. Traffic that does not fall into this profile will be deemed anomalous and should be reviewed.

A key point here can be made about what other traffic should be allowed or may be needed for troubleshooting. That is where the tuning of the device comes into play as it does with any IDS. You will need to run it for a period of time. You should adjust it for ICMP, SSH, and other minor things you need to manage the servers. Once you have it tweaked, you should have a normal profile or baseline of the traffic you are expecting by eliminating the false positives.

The anomalous sensor is now looking for any traffic that does not match this profile, and it can alert you as you have it configured. I will go into more examples later of attacks and suspicious traffic that will be found like this as I explain the tool I use. This is a brief explanation of how anomalous detection in general works against a network baseline of an Internet connection. This gets increasing more complex as you add more clients with diverse protocols and higher bandwidth.

One such appliance designed for this type of detection is StealthWatch by Lancope. This device is an upcoming accessory in the ID market. It is an appliance that helps you create a profile of your network based on types of traffic and bandwidth utilization. Then it alerts you to deviation from the profile and abnormalities. StealthWatch is a fairly new technology that is helping catch intrusions from a different perspective. In many ways, it helps you validate your corporate policies by showing you exactly what is happening on your network.

General:

StealthWatch is based on a hardened linux kernel with a small footprint. It comes as an appliance with custom network device drivers to allow it to handle up to 1GB of bandwidth. The device has a small form factor and is rack-mountable. It comes with two network ports, one for management and one for monitoring. SSH is used for command line access and SSL for the web interface. It uses "data flow analysis" to determine irregularities in network traffic.

Profiling:

After the box is setup and plugged in, it begins to profile your network. It has several different modes that you step through in order to let it configure itself to your network. It generates a Service Profile of all the IP addresses it sees from monitoring the traffic. The profile allows you to see what is really happening on your network. Once you have the profile built correctly through tuning, you can lock it in. This profile is the baseline for your network segment. Anything not adhering to the profile causes alerts. The non-profile traffic increases the Content Index (covered in the next section) of each data flow.

What about new legitimate traffic? StealthWatch has a way of allowing you to edit the Service Profile. In the case of putting up a new web server, the appliance would alert you to new data flows outside of the current profile. You would then have the option to accept them as legitimate traffic, thus adapting your profile to your changing environment.

Bandwidth also affects Content Index. This is one of the ways StealthWatch can detect DoS attacks or suspicious traffic. This is where the real anomalous detection comes into play. Valid traffic can be abnormal if it consumes too much bandwidth. StealthWatch will watch for abnormal amounts of traffic while reporting on how much of your bandwidth is being used. A lot of worm traffic is caught due to the nature of its attack method. Many use higher bandwidth and random IP selection.

Concern Index (CI):

The Concern Index is a number that is generated for each IP data flow based on the type of traffic. A threshold is set on the box for CI, and once a data flow exceeds the threshold, an alert is generated. StealthWatch uses a sophisticated algorithm to determine this number based on the traffic, and it helps StealthWatch determine anomalous traffic. Different factors affect this number such as malformed packets, high bandwidth (bytes transferred), and the Service Profile.

Once you figure out where your threshold is, you will want to be suspicious of anything that crosses it. Even if it is now added to the existing profile, you will want to verify that it was legitimate traffic. The key to this tool is that it can alarm you to changes in your traffic that you won't normally see. You would be surprised how important that information can be.

Display:

The home page of the StealthWatch appliance is like an operations center. It shows you CI and bandwidth statistics in graph form. It also has graphs for the traffic flow ingress and egress over the past hour. You can easily determine if you have any alarms. All normal traffic is in green and all alarms are in red. All the graphs and statistics on this page are hyperlinks to more detailed analysis. There is a menu on the left side for more options too. It is a simple, secure web interface, which refreshes itself periodically (every minute I believe). There are icons for the advanced configuration options, graphing, and printing options. It is like having all the survival tools on one console page. Overall, the StealthWatch display is easy to read and easy to determine when something is wrong.

Alerting:

There are several types of alarms in StealthWatch. Minor alarms are triggered for the CI abusers. Hosts or data flows that violate the CI threshold are considered "Host High Concern" for data irregularities and "Host High Traffic" for bandwidth violations. The major alarms are triggered if something has already triggered a minor alarm and then also sends data across the data flow. It can alert by email and SNMP notification.

The box also does some reconnaissance for you. It will do a host name lookup of the attacker, a traceroute to his box, and an ARIN lookup for the owner of the IP address.

Examples:

Any time these is a major irregularity in network traffic, an alarm is generated. Suppose there was an internal email problem where a server failed for some reason and smtp mail queues that needs to be sent to the Internet. Then, after the problem is resolved, the email server has to relay out a large amount of backed up email. This will

trip the StealthWatch box because of the growth in outgoing email traffic. This is normal traffic in an abnormal situation.

Nimda and CodeRed are malicious Internet worms that attack over several well-known ports. Due to the high traffic volume of these worms, StealthWatch is adept at alerting when particular IP addresses are constantly banging on your door. This is especially valuable if you do not have the latest patches of virus update on your server. The appliance will help catch unknown or new attacks due to the nature in which they spread.

Scanning and probing is a common way hackers first discover and perform reconnaissance. With updated servers, a good border router, and a tight firewall, you can reduce the information a hacker can receive. However, he will still try his tricks. The key here is when he launches an exploit. StealthWatch builds a CI for each "data flow". If the hacker continues to probe, a minor alarm for high activity is sent. If the hacker runs an exploit, transferring data, then a major alarm is sent. I have seen this first hand but in an accidental fashion. One of our security professionals was probing our Internet facing servers from his home. After a while, he tripped the minor alarm which I was expecting since it was a security test. However, when he logged into his Internet email account, which we host, to send the data to me, I immediately was caught off guard to get a major alert confirming an apparent security breach. This was a false positive of course, but it proved that the StealthWatch product was closely watching the traffic coming from the IP of my coworker.

In the end, there is no complete IDS solution. There are layers of defense. You can add and add layers until you are comfortable that you have reduced your risk level to an acceptable level. Behavioral IDS has its issues and is just another way of confronting the ways to catch malicious traffic. It seems the more technology you can leverage, the better chance you have of catching your opponent. With budgets and limiting spending in an area that sometimes has trouble showing Return On Investment, you can't always have everything you want. StealthWatch is an upcoming technology that will help close the gap on monitoring all that traffic and giving Intrusion Analysts another way to look at the traffic. As the technology matures so do the hackers. We are constantly challenged to think outside the box and analyze things differently.

References:

1. Debar, Herve. "What is behavior-based Intrusion Detection?" SANS Institute Online Reading Room. URL: http://www.sans.org/newlook/resources/IDFAQ/behavior_based.htm (8 Feb. 2002)

2. NorthCutt, Stephen and Novak, Judy. Network Intrusion Detection Handbook An Analyst's Handbook. Second Edition; Indianapolis:New Riders, September 2000.

3. Lancope, Inc. URL: http://www.lancope.com. (5 Feb. 2002)

4. Chu, Francis. "Lancope IDS Looks Into the Unknown to Detect Threats." Eweek.com. URL: http://www.eweek.com/article/0,3658,s%3D702&a%3D18030,00.asp (8 Feb. 2002)

5. Bankster, Jeff. "StealthWatch Intrusion Detection Appliance." SCMagazine Online. URL: http://www.scmagazine.com/scmagazine/2001_08/review/review1.html (8 Feb. 2002)

6. Yasin, Rutrell. "New Way to Detect Hacking." Internetweek Online. URL: http://www.internetweek.com/infrastructure01/infra062501-3.htm (8 Feb. 2002)

7. Tanase, Matthew. "The Future of IDS." SecurityFocus.com. URL: http://www.securityfocus.com/infocus/1518 (8 Feb. 2002)

Assignment 2 -- Network Detects

Detect 1

```
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.72:4000 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.77:4001 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.77:4002 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.77:4003 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.104:4000 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.104:4001 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.104:4002 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.104:4003 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.100:4002 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.100:4003 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.102:4000 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.102:4001 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.102:4002 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.102:4003 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.103:4000 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.103:4001 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.103:4002 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.103:4003 UDP
```

1. Source of Trace:

This trace was detected by an IDS server placed on the public side of my employer's Internet connection.

2. Detect was generated by:

The detect was generated by a Snort IDS running on RedHat Linux 7.1. I have removed the network segment address by obscuring it with "x.x.x". The logging format is straightforward. Each line represents one packet. Starting from the left, the fields are date, timestamp, source IP:source port, direction of packet, destination IP:destination port, and finally protocol.

3. Probability the source address was spoofed:

There is a low probability that the address was spoofed. The trace is a collection of packets targeting consecutive IP addresses within a subnet. The attacker most likely was looking for certain services or IP addresses to respond to these packets. Therefore, since a response was warranted and the packets do not follow a DoS pattern, they are most likely not spoofed.

The IP address of the attack comes from China. (http://www.apnic.net/)

inetnum	61.139.0.0 - 61.139.127.255
netname	CHINANET-SC
descr	CHINANET Sichuan province network
descr	Data Communication Division
descr	China Telecom
country	CN
admin-c	CH93-AP, inverse
tech-c	XS16-AP, inverse
mnt-by	MAINT-CHINANET, inverse
mnt-lower	MAINT-CHINANET-SC, inverse
changed	hostmaster@ns.chinanet.cn.net 20000601
source	APNIC

4. Description of attack:

The attack is a scripted or deliberate search of IP addresses on a subnet with UDP packets. The attacker is most likely looking for ICQ services running on a server, and at the same time mapping the network by looking for servers that will respond. The attack tries ip addresses on the same subnet, and since the timeframe of the packets is short, this is most likely a scripted attack if not a tool running through a list of IP addresses.

5. Attack mechanism:

Since the attack changes IP addresses and stays within the same UDP range, it is looking for responses to gather information about servers for sure and maybe even ports.

The attacker produces UDP packets with the same source port that scans a range of IP addresses for responses on UDP ports ranging from 4000-4003. Since there was no evidence of an exploit being used, it initially looks like the attacker was still trying to discover vulnerable servers by looking for services running on these high UDP ports. Due to the speed of the packets, the attack was launched from a tool or a script. I have not located the exact tool that made this scan. The static source port is a key to whatever generated this attack.

6. Correlations:

The information below came from incidents.org's website.

http://www.incidents.org/diary/november01/111201.php#3

==========

Scans to UDP Ports 4000, 4001, 4002, and 4003

Throughout the past month there has been a significant
amount of scanning activity to ports 4000/udp - 4003/udp.
The probing is always sourced from addresses in China,
and typically an attacker scans for several of the ports
at the same time.

An overview of recent activity is given below. The numbers
listed under the port columns (labeled 4000, 4001, etc.)
are the number of targets probed on the specified port by
the attacking IP.

Date	SourceIP	4000	4001	4002	4003
10-23	61.182.241.77	64	1	2	1
10-25	202.110.163.108	20	20	20	19
10-25	61.134.228.232			1	18
10-26	61.167.249.201	233	233	233	248
10-27	211.97.183.67	425	238	215	200
10-28	61.182.251.89	31	13	8	16
10-29	61.182.40.85	160	159	167	159
10-30	202.111.161.129	12	9	10	13
11-01	61.184.166.11	255	190	74	75

11-04 61.180.215.2 246 246 245 245
11-09 61.156.112.13 23
11-09 61.180.188.54 162 164 164 159
11-10 210.51.226.250 121 134 124 129

As an example, a few log recent log excerpts are included below.
Note that in all cases the probes are generated very rapidly
and the source port is held constant in the scan.

2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.255 - 4001 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.254 - 4002 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.247 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.249 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.241 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.239 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.239 - 4001 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.237 - 4001 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.230 - 4000 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.223 - 4002 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.222 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.217 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.216 - 4000 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.206 - 4002 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.208 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.199 - 4000 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.198 - 4002 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.197 - 4000 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.193 - 4003 - UDP
2001-11-11 17:21:57 +0100 210.51.226.250 - 11851 - 262.109.129.191 - 4000 - UDP

NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.131:4001 L=35 S=0X00 I=56891 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.131:4000 L=35 S=0X00 I=56635 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.130:4003 L=35 S=0X00 I=56379 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.130:4002 L=35 S=0X00 I=56123 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.130:4001 L=35 S=0X00 I=55867 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.130:4000 L=35 S=0X00 I=55611 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.129:4003 L=35 S=0X00 I=55355 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.129:4002 L=35 S=0X00 I=55099 F=0X0000 T=49

NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.128:4003 L=35 S=0X00 I=54331 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.129:4001 L=35 S=0X00 I=54843 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.128:4002 L=35 S=0X00 I=54075 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.129:4000 L=35 S=0X00 I=54587 F=0X0000 T=49
NOV 10 04:27:10 PROTO=17 61.180.188.54:1044 273.47.17.128:4001 L=35 S=0X00 I=53819 F=0X0000 T=49

2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.132.121 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.132.189 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.132.46 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.132.88 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.132.24 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.129.210 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.129.249 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.129.167 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.129.119 - 4000 - UDP
2001-11-10 07:32:56 +0100 61.156.112.13 - 1958 - 262.109.129.71 - 4000 - UDP

2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.156 - 4000 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.155 - 4003 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.155 - 4002 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.155 - 4001 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.155 - 4000 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.154 - 4003 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.154 - 4002 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.154 - 4001 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.154 - 4000 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.153 - 4003 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.153 - 4002 - UDP
2001-11-04 04:58:25 -06:00 61.180.215.2 - 1430 - 192.168.64.153 - 4001 - UDP

Russel Fulton reported similar activity about a year ago, but the news group thread did not reach a conclusion regarding what the attackers were looking for.
http://archives.neohapsis.com/archives/incidents/2000-11/0189.html

It is possible that the scan is intended to ferret out ICQ servers, which listen on port 4000/udp. The page linked below provides a number of ICQ-based attacks and exploits:

http://the-hack.net/icq/

At this point it is still unclear what the attackers are looking for however.

These examples correlate with my findings. The static source port is again displayed in these detects. There is no definitive answer to what this scan is used for. The best guess is still ICQ services for exploit. Another interesting fact is that the 61.x.x.x range of networks is registered in China. This seems to be where a large majority of the attacks originate by looking at the traces above in the Incidents.org information.

7. Evidence of active targeting:

Since this attack spreads out over several IP addresses, the attacker is most likely involved in reconnaissance. This could be a warning for more attacks and probing in the future. At this time, there was no threat of vulnerability exposure due to the nature of the scan. The IP address was not found to have selected a target for further probing.

8. Severity:

(Critical + Lethal) – (System Countermeasures + Network Countermeasures) = Severity

Criticality: 3 – User is looking for ICQ server or network mapping.
Lethality: 2 – Attacker did not use exploit and still in discovery mode.
System Countermeasures: 5 – All servers modern OS with patches.
Network Countermeasures: 5 – Restricted firewall in place disallowing UDP.

Severity: (3 + 2) – (5 + 5) = -5

9. Defensive recommendation:

A firewall is currently in place on this segment, and these UDP ports are blocked from the Internet facing servers. I also recommend watching for any future probes that are similar for correlation purposes. A couple rules to the firewall and IDS would help to provide more data in the future should the attacker return. At this time, the firewall protects this segment from this type of attack.

10. Multiple choice test question:

Which fields are most suspicious in the following packets?

```
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.72:4000 UDP
Dec  7 17:32:13 61.139.82.132:1079 -> x.x.x.104:4000 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.102:4000 UDP
Dec  7 17:32:14 61.139.82.132:1079 -> x.x.x.103:4000 UDP
```

A. time and date

B. source IP address and destination IP address
C. source port, destination port, and timestamp (**CORRECT**)
D. protocol and timestamp

Detect 2

```
Dec 11 15:39:39 192.168.1.100:2768 -> x.x.x.220:80 FIN *******F
Dec 11 15:39:42 192.168.1.100:2769 -> x.x.x.220:80 VECNA *2U*P***
Dec 11 15:39:42 192.168.1.100:2769 -> x.x.x.220:80 NOACK *2**PR*F
Dec 11 15:39:38 66.130.166.107:2769 -> x.x.x.220:80 SYN ******S*
Dec 11 15:39:40 66.130.166.107:2766 -> x.x.x.220:80 XMAS **U*P**F
Dec 11 15:39:40 66.130.166.107:2769 -> x.x.x.220:80 NULL ********
Dec 11 15:41:09 66.130.166.107:2770 -> x.x.x.220:80 SYN ******S*
Dec 11 15:41:10 66.130.166.107:0 -> x.x.x.220:2769 NOACK *2**PRSF
Dec 11 15:41:10 192.168.1.100:2768 -> x.x.x.220:80 INVALIDACK *2*A*R*F
Dec 11 15:41:11 192.168.1.100:2770 -> x.x.x.220:80 VECNA *2**P***
Dec 11 15:42:13 192.168.1.100:1 -> x.x.x.220:2768 INVALIDACK *2UA**SF
Dec 11 15:42:15 192.168.1.100:2768 -> x.x.x.220:80 NOACK 1*U*PR*F
Dec 11 15:42:16 192.168.1.100:2770 -> x.x.x.220:80 SYNFIN ******SF
Dec 11 15:42:14 66.130.166.107:0 -> x.x.x.220:2770 SPAU 1*UAP*S*
Dec 11 15:42:15 66.130.166.107:0 -> x.x.x.220:2770 NOACK 1*U*PR**
Dec 11 15:42:18 192.168.1.100:2770 -> x.x.x.220:80 INVALIDACK 1**A*RS*
Dec 11 15:42:19 192.168.1.100:2770 -> x.x.x.220:80 NOACK 12U***S*
Dec 11 15:42:20 66.130.166.107:2770 -> x.x.x.220:80 NULL ********
Dec 11 15:42:21 66.130.166.107:2770 -> x.x.x.220:80 NOACK 12**P*S*
Dec 11 15:42:23 192.168.1.100:134 -> x.x.x.220:2768 SYNFIN 12****SF

Dec 14 23:51:53 192.168.1.100:1082 -> x.x.x.220:80 INVALIDACK 1*UAP*SF
Dec 14 23:51:56 192.168.1.100:1081 -> x.x.x.220:80 NOACK 12**P*S*
Dec 14 23:51:51 66.130.166.107:1084 -> x.x.x.220:80 SYN ******S*
Dec 14 23:51:55 66.130.166.107:1084 -> x.x.x.220:80 NOACK 12***RS*
Dec 14 23:52:24 192.168.1.100:1082 -> x.x.x.220:80 VECNA 12U*****
Dec 14 23:52:45 66.130.166.107:1081 -> x.x.x.220:80 NOACK 12U*PRSF
Dec 14 23:52:48 66.130.166.107:0 -> x.x.x.220:1081 INVALIDACK 12UAPR*F
Dec 14 23:52:47 192.168.1.100:1083 -> x.x.x.220:80 NOACK **U***S*
Dec 14 23:52:52 66.130.166.107:1082 -> x.x.x.220:80 INVALIDACK **UA**SF
Dec 14 23:52:50 192.168.1.100:1083 -> x.x.x.220:80 SPAU 12UAP*S*
Dec 14 23:52:51 192.168.1.100:1082 -> x.x.x.220:80 NOACK 12U***S*
```

1. Source of Trace:

This trace was detected by an IDS server placed on the public side of my employer's Internet connection.

2. Detect was generated by:

The detect was generated by a Snort IDS running on RedHat Linux 7.1. I have removed the network segment address by obscuring it with "x.x.x". The logging format is straightforward. Each line represents one packet. Starting from the left, the fields are date, timestamp, source IP:source port, direction of packet, destination IP:destination port, and finally protocol.

3. Probability the source address was spoofed:

There is some discussion here. Since the packets are from two different addresses one on of them was a private address, there is a high probability that those packets had a spoofed source address. The attacker could not guarantee that those packets would return, and since he continued with the attack at two separate times, there is evidence there to conclude that there was spoofing involved. The 192.168.0.0/16 is a private address range that is not routed on the Internet.

The other address (66.130.166.107) is most likely not spoofed. This is the other address in this trace, and it is a routable address on the Internet. Since the attack looks like a reconnaissance attack, this address most likely is not spoofed for information gathering purposes. There is a strong possibility that the private address was used to help mask the attack or try to evade IDS.

4. Description of attack:

The trace is a collection of packets with invalid and abnormal flags set that have been sent from multiple IP addresses to an Internet web server. The attacker was using common responses to invalid stimuli to determine the operating system of the targeted server. There is a mix of source IP address which seems very relevant and a possibility of spoofing to further complicate the attack. This is a discovery phase.

5. Attack mechanism:

Since the packets are abnormal, there has to be a tool crafting these packets. Many of the packets in this trace would not be found on a normal network. They have obviously been forged to take advantage of different replies to different stimuli. There has been quite a mixing of TCP flags with reserved bits as well. In the two examples, the IP addresses are consistent and the source ports stay within a four port range. Since the privately addressed packets could not guarantee a response, there looks to be some evasion technique involved such as the decoy option with NMAP, or rather, there was a misconfiguration of the tool on the attacker's part. There is a small chance that those packets are irrelevant. Since they appear on different days and mimic similar behavior, there is a greater likelihood of clouding the attack.

6. Correlations:

When I first saw this scan, I thought there were two different hosts involved, and that there attacks just overlapped. On further analysis and correlation, I found a similar scan three days later with a similar signature. These traces matched up in design and behavior. I searched for a similar trace, but could not find an exact match. There are plenty of examples of OS fingerprinting. This is one of the most unique I have seen. NMAP has various scans that could cover some of the packets in this trace. However, even with the decoy option, the advanced packet forging here leads me to conclude that there was a different, more powerful tool involved. Also, I was unable to find a good example of combining spoofed packets with valid packets during an OS fingerprint.

I did not think this was a port scan, but rather a more in-depth scan of one open port to determine the underlying operating system.

The firewall plays an interesting part in this attack too. After researching the stateful capabilities of the firewall, I have found that most of the packets would not have made it through. Since the policy is very strict about source and destination packets, the private ones would never have made it to the web server. Also, the conflicting ports and flags would cause most to be dropped being that they are no part of an existing connection or the same connection. This attack was crafty, but not quite as successful as I thought. This could explain why the attacker returned three days later. It might have been to supply more information for the fingerprint, especially if the first trace was a scripted attack.

7. Evidence of active targeting:

This was a strong attempt by an attacker to determine the OS of the web server. There is a high probability of active targeting due to the nature of the reconnaissance and depth of the attack. There was a likely attempt at evasion. The attacks were also short and sweet (informative). This attacker should be watched very cautiously for further probing and possible exploit attempts.

8. Severity:

(Critical + Lethal) – (System Countermeasures + Network Countermeasures) = Severity

Criticality: 5 – Internet web server was targeted.
Lethality: 2 – Attacker did not use exploit and still in discovery mode.
System Countermeasures: 2 – Modern OS with patches, but packets out of spec.
Network Countermeasures: 3 – Stateful Firewall with port open though.

Severity: (5 + 2) – (2 + 3) = 2

9. Defensive recommendation:

There is already a firewall protecting the web server. However, port 80 is allowed for web access, and the nature of the attack solicits information over allowed ports. This attack can be successful with the existing setup. It is important to monitor the attacker's behavior and IP address. There is a strong probability that this attacker may return to try an exploit on the system probed. We would want to be ready and prepared to respond to further actions by this attacker.

10. Multiple choice test question:

Which set of flags below is valid for a TCP packet?

A. FIN & ACK (**CORRECT**)
B. SYN & RST

C. SYN & FIN
D. SYN, PSH, URG, & RST

Detect 3

16:57:05.581863 4.4.78.59.2843 > x.x.x.19.20034: S [tcp sum ok] 8653933:8653933(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 14219, len 48)
16:57:10.829864 4.4.78.59.3097 > x.x.x.19.1243: S [tcp sum ok] 8659208:8659208(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 40844, len 48)
16:57:16.116863 4.4.78.59.3353 > x.x.x.19.30100: S [tcp sum ok] 8664429:8664429(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 7822, len 48)
16:57:21.414863 4.4.78.59.3607 > x.x.x.19.6670: S [tcp sum ok] 8669711:8669711(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 38031, len 48)
16:57:26.692863 4.4.78.59.3862 > x.x.x.19.2583: S [tcp sum ok] 8674941:8674941(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 5521, len 48)
16:57:31.939864 4.4.78.59.4117 > x.x.x.19.1016: S [tcp sum ok] 8680272:8680272(0) win 8192 <mss 536,nop,nop,sackOK> (DF) (ttl 114, id 31634, len 48)

1. Source of Trace:

This trace was taken from a computer at my home that is directly on the Internet through a cable modem.

2. Detect was generated by:

The detect was generated by BlackIce running on a Windows NT 4.0 server. The trace file that BlackIce provides for logging was in raw format. I read it into Ethereal for analysis, then output it again in tcpdump format. I then ran the file through windump to get the text trace of the packets shown here. The destination addresses have been sanitized as well.

The log format is timestamp, source IP.port, destination IP.port, tcp flags set, checksum, sequence numbers, window size, TCP options, fragment flag, time-to-live, id field, and length.

3. Probability the source address was spoofed:

There is a low probability that the address was spoofed. The attacker seems to be looking for open Trojan ports by a TCP scan of hosts. For this attack to be successful, the attacker would need to receive packets confirming the open ports. It is most likely the source address is not spoofed.

4. Description of attack:

The attacker scans a list of well-known Trojan TCP ports looking for any response that would suggest a host has been compromised. Initially, it looks like a SYN packet to random TCP ports every 5 seconds. However, the ports probed are very

special. It is a loud and quick scan for possible Trojan Horse code running on infected machines.

5. Attack mechanism:

The trace shows six SYN packets sent to random ports. By looking at the following url (http://www.sans.org/newlook/resources/IDFAQ/oddports.htm), I was able to identify the Trojan ports probed.
Here are the suspicious ports:

port 20034 NetBus 2.0 Pro, NetRex, Whack Job
port 1243 BackDoor-G, SubSeven , SubSeven Apocalypse, Tiles
port 30100 NetSphere
port 6670 BackWeb Server, Deep Throat, Foreplay or Reduced Foreplay, WinNuke eXtreame
port 2583 WinCrash
port 1016 Doly Trojan

The attacker is using a simple TCP port scan to look for Trojan Horse programs that may have infected machines. The packets are all TCP SYN packets with incrementing initial sequence numbers, same window size, same TTL value, same TCP options, and incrementing source port. The source port steps at large intervals. The intervals are 254, 256, 254, 255, and 255 respectively. There seems to be something fishy here. This could be a scripted attack over a range of servers. The small change between the numbers may have to do with responses to the scan. It does seem like a revolving scan if it returns to the same IP address about every 255 source ports. It is most likely scanning a whole segment. There seemed to be no attempt to hide or mask this attack.

If the attacker receives a response, there is a better than likely chance of an infected host. Trojan Horse programs are malicious code installed on a client machine that allows an attacker to take control or use the system resources of that computer whenever needed. Most of the time the user is unaware of this activity. In this case the trace shows of an attempt to access some of the more common Trojans such as NetBus, SubSeven, Deep Throat , and Wincrash.

6. Correlations:

From Nelson Carter's GCIA Practical (0374), there is a good description of what SubSeven is and how it works:

SubSeven is a remote access and control program that can perform a wide variety of functions ranging from common annoyances to full blown compromise of files and critical data on the infected host(s). Since its creation by a hacker who calls himself Mobman, SubSeven has gone through many version changes and now is at its latest release: Version2.2. There are three main components to this new version; the server, the edit server program, and the client. In order for this all to work, first the hacker must install the server on the host(s) of choice, this is commonly done through e-mail file attachments. This server file can be customized through the use of the Edit Server

program, in which the hacker can change many parameters such as server port number, installation methods, executable name, methods of notifying the hacker that this host is online and many others (described in detail in "The Components" section). Once the server portion is successfully installed on the host the hacker can now use the SubSeven client to attach to the host via the port that was pre determined by the Edit Sever program (27374 by default). Now the fun begins, the hacker can perform many tasks on the victim host ranging from changing hardware settings, (changing window colors, opening and closing the cd-rom, reversing the mouse buttons, rebooting the computer, etc.), to information gathering, (windows version, user's name and address, hard drive and file information, and password information) and installing program updates to the infected host. The SubSeven sever host is not the only victim here the hackers can also scan other networks and perform DDoS attacks from the infected host, all the while keeping his identity and involvement a secret. New features in version 2.2 include support for socks proxies, a packet sniffer, random port listening (notifications of changes are sent to hacker), CGI notifications and the ability to send keystrokes to remote system(s)

He (Nelson Carter) gave an example trace of a SubSeven scan:

```
Feb 16 22:03:47 141.150.211.31:1920 -> x.x.x.2:27374 SYN ******S*
Feb 16 22:03:47 141.150.211.31:1921 - > x.x.x.3:27374 SYN ******S*
Feb 16 22:03:47 141.150.211.31:1922 -> x.x.x.4:27374 SYN ******S*
Feb 16 22:03:47 141.150.211.31:1923 -> x.x.x.5:27374 SYN ******S*
Feb 16 22:03:47 141.150.211.31:1925 -> x.x.x.7:27374 SYN ******S*
Feb 16 22:03:47 141.150.211.31:1929 -> x.x.x.11:27374 SYN ******S*
```

This scan shows similar behavior to the one I detected, but it is specifically for SubSeven. TCP SYN packets directed at the listening port of the Trojan Horse. Port 27374 is another well-known port that a different version of SubSeven listens on. The trace here is searching IP addresses by port to find a compromised host.

Here is a description of NetBus 2.x Pro by Internet Security Systems in an alert they issued (http://xforce.iss.net/alerts/advise20.php):

ISS Vulnerability Alert
February 19, 1999

Windows Backdoors Update II:
NetBus 2.0 Pro, Caligula, and Picture.exe

Synopsis:

This advisory is a quarterly update on backdoors for the Windows 9x and Windows NT operating systems. The focus of this advisory is NetBus 2.0 Pro. The final version of NetBus 2.0 Pro was released on February 19. The new version of NetBus is not distributed as a backdoor, but as a "Remote Administration and Spy Tool." Due to the proliferation of NetBus and its common use in attacks across the Internet, NetBus 2.0 poses a significant risk with its new functionality and enhanced network communication obfuscation. The default installation of NetBus 2.0 Pro (NB2) does not hide itself from the user, but it does support an

"Invisible Mode" to prevent users of infected machines from noticing the software. The version of NB2 available on the Internet notifies users upon installation, however attackers can easily hide the installation with slight modification.

This ISS X-Force Security Alert also includes information about the Picture.exe trojan and the Caligula macro virus, since the presence of either of those on your system could lead to a compromise of security and transmission of sensitive data over the Internet.

NetBus 2.0 Pro Description:

NB2 includes enhanced functionality, including the ability to find cached passwords, full control over all windows, capturing video from a video input device, a scheduler to run scripts on specified hosts at a certain time, and support for plugins. Plugins will enable programmers at add functionality to NB2, similar to the architecture provided in the cDc BackOrifice backdoor. The only plugin currently available is a file-finding utility that searches a victim's hard drive for files.

By default, NB2 listens on TCP port 20034, but this is easily configurable. NB2 uses a weak form of encryption to obfuscate its communications, but the format of its packets makes it easy to spot NB2 traffic. Each packet starts with 'BN', followed by the following sequence:

- - - - Two bytes representing the length of the packet.
- - - - Two bytes of 0x02 or 0x00, probably for the version of NetBus.
- - - - Two random bytes, probably to confuse people.
- - - - Two bytes for the command code.

For example:

42 4E XX XX 02 00 YY YY ZZ ZZ ...data...

XX XX is the length of the whole NetBus 2.0 packet
YY YY are just two random bytes
ZZ ZZ is the command code

The first 2 bytes are 'BN', the length of the packet is XX XX, and the version is 0x02.

NB2 stores registry information in the HKEY_CURRENT_USER\NetBus Server registry key. If you have this key in your registry, NB2 may be running on your machine. To determine the port that NB2 uses, check the value of HKEY_CURRENT_USER\NetBus Server\General\TCPPort, and use the 'netstat -an | find "LISTEN"' command to see if your system is listening on that port. If NB2 is listening, you need to find the NB2 server executable and delete it. The default name is NbSvr.exe, but it can be easily renamed.

If NetBus 2.0 is configured to start automatically when your computer boots, the HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices registry key will have a registry value called 'NetBus Server Pro' that specifies the full path for the location of the NetBus executable. Use the registry key value to locate and delete the file if you find that NB2 has been installed on your machine without permission.

NetBus 2.0 traffic using the default port can be detected by RealSecure if you configure it to monitor traffic on TCP port 20034.

This is another example of a program that can be used as a Trojan Horse. This is a more recent version of NetBus. The use of this program should be monitored and securely contained. The corresponding CVE "under review" is CAN-1999-0660.

Also, Tadaaki Nagao in his GCIA Practical (0187) identifies this type of scan as a Multiscan and has examples of traces confirming my source port suspicions.

7. Evidence of active targeting:

This trace looks like a random scan on the Internet for miscellaneous Trojans. The attacker was trying to find a vulnerable host by scanning TCP ports. Since my computer was scanned and no information returned to the attacker, there is no reason for my computer to be actively targeted. Also, the source IP address did not return in my logs. The attacker targeted my subnet but not my computer.

8. Severity:

(Critical + Lethal) – (System Countermeasures + Network Countermeasures) = Severity

Criticality: 3 – Home computer was attacked directly on Internet.
Lethality: 4 – Probing for multiple Trojans on compromised hosts.
System Countermeasures: 5 – NT System with latest patches and virus protection
Network Countermeasures: 5 – Host firewall (BlackICE)

Severity: (3 + 4) – (5 + 5) = -3

9. Defensive recommendation:

Since this scan involves non-standard ports, a good host firewall or IDS will catch this access. Also, most modern and updated virus protectors will detect rogue code that falls under Trojan or malicious executables. Good procedures for email usage, keeping virus protection updated, and alerting on non-standard port access will help prevent unwanted Trojan access.

10. Multiple choice test question:

In this trace, source port numbers that are about 255 numbers apart in consecutive connections most likely indicates what?

A. The packets are being crafted.
B. The attacker is scanning a Class C subnet. (**CORRECT**)
C. It is irrelevant in this trace.
D. Nmap is being used to scan this host.

Detect 4

00:10:52.386175 206.135.164.221.2945 > x.x.x.124.3128: S [tcp sum ok] 2087543395:2087543395(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) (ttl 112, id 50919, len 48)
00:10:52.386175 206.135.164.221.2946 > x.x.x.124.8080: S [tcp sum ok] 2087597088:2087597088(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) (ttl 112, id 50920, len 48)
00:10:55.631175 206.135.164.221.2945 > x.x.x.124.3128: S [tcp sum ok] 2087543395:2087543395(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) (ttl 112, id 51180, len 48)
00:10:55.631175 206.135.164.221.2946 > x.x.x.124.8080: S [tcp sum ok] 2087597088:2087597088(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) (ttl 112, id 51181, len 48)

1. Source of Trace:

This trace was taken from a computer at my home that is directly on the Internet through a cable modem.

2. Detect was generated by:

The detect was generated by BlackIce running on a Windows NT 4.0 server. The trace file that BlackIce provides for logging was in raw format. I read it into Ethereal for analysis, then output it again in tcpdump format. I then ran the file through windump to get the text trace of the packets shown here. The destination addresses have been sanitized as well.

The log format is timestamp, source IP.port, destination IP.port, tcp flags set, checksum, sequence numbers, window size, TCP options, fragment flag, time-to-live, id field, and length.

3. Probability the source address was spoofed:

The attacker in this trace is searching for specific open ports on machines. For this information to get back to the attacker, the TCP 3-way handshake would have to complete or data (return packets) would need to be received from the destination. Therefore, the source address is most likely not spoofed.

4. Description of attack:

The trace shows two separate attempts of packets to enumerate open ports related to possible proxy services on the host machine. Similar to scanning for Trojans, this is a most likely a scan for proxy servers, specifically a squid proxy (3128). The intent is to locate an infected host for further exploit. RingZero Trojans are also known to use these ports and have a similar signature.

5. Attack mechanism:

The attacker seems to be scanning for the ports 8080 and 3128, which are known proxy listening ports. The packets are interesting too. They look like retries for the most part in that the ports are tried in succession with the same initial sequence numbers. Also, the source ports stay the same on the connection attempts respectively. However, the id field has not incremented the same. There is a significant jump from packets 3 and 4. The difference is a value if 260. This would most likely indicate that the host was doing something else while these packets were sent to me. However, the timeframe is about 3 seconds according to the timestamp. A strong case for scanning or automation could be made here. This would further incur that this is a search over many addresses for hosts that offer proxy services.

There are some important exceptions here. If you look at a RingZero signature, you usually see packets to port 80. There are no packets to port 80 here. According to the SANS conference, RingZero traces usually contain packets destined to port 80, 8080, and 3128. However, we can not rule this scan out without further information. It could possibly be a RingZero scan crafted to look like a less suspicious proxy service scan.

6. Correlations:

There is not enough information here to draw a strong conclusion to what the attacker is doing. The attack is definitely a scan on ports that represent most likely a proxy search. Not much seems to have been done to obscure the attack. More information would be needed to evaluate this attack better and to rule out a RingZero scan attack or some type of network reconnaissance.

Since the source IP address comes from a cable network in Chicago, the scan gets more complicated.

From www.arin.net:

Epoch Networks (NETBLK-HLC-3-EPOCH) HLC-3-EPOCH 206.135.0.0 - 206.135.255.255

Prime Cable Of Chicago (NETBLK-EPOCH-1692) EPOCH-1692 206.135.164.0 - 206.135.164.255

Here is some more info from a similar trace from incidents.org (http://www.incidents.org/archives/y2k/122999-1630.htm):

December 29, 1999 1630

One contributor has reported over 2200 log entries demonstrating alternating probes against ports 3659 and 3670 starting at 12/28/1999 22:16:41.258 and ending at 12/29/1999 11:54:07.084 –0500 GMT (Eastern Time Zone)

Analysis: Could be something new or a misconfigured application on the offending network. Given the scanner's boldness to continue the activity for ~13 hours, I'm hopeful the issue is one of misconfiguration.

I've seen very few messages to handler@incidents.org this week with activity in the 3000's and these ports aren't showing up on know threat lists. Any log entries correlating this contributor's observations would be

helpful.

Our caller is either stabbing in the dark or looking for something he suspects is there.

19991228 22:19:05 1999 172.20.104.1-->10.0.0.62 src port 3757, dest port:3128, protocol:6.
19991228 22:19:05 1999 172.20.104.1-->10.0.0.62 src port 3756, dest port:8080, protocol:6.
{pattern continues…}.

19991228 22:19:03 1999 172.20.104.1-->10.0.0.2 src port 3636, dest port:8080, protocol:6.
19991228 22:19:03 1999 172.20.104.1-->10.0.0.2 src port 3637, dest port:3128, protocol:6.

Analysis: First note an extended discussion on proxies may be found at http://www.sans.org/y2k/proxy.htm.

The offender has stacked up requests on his outgoing ports 3636 through 3757 – a range of 121 ports. The offender uses one port to probe port 8080, then a second to probe port 3128 on the first IP address in this range of 61 addresses. We can tell from the logs that it's a single application being used to perform the port scan; even with the logs demonstrate that the packets were received out of order – most likely due to congestion, the next IP address returns to the 8080 first, 3128 second pattern.

You may recall the RingZero trojan, which probed ports 80, 8080 then 3128, then attempted to FTP information to a xx.yy.RU domain name. It's a hunch, but this attack originated from a xx.yy.RU domain. Coincidence? If you're not familiar with RingZero, more information is available at: http://www.sans.org/audioplay/ringzero, and is worth reading to understand that Trojan's behavior and implications.

Current indications are that the FTP server used in RingZero is back online at 193.86.194.77. Reviewing my handler@incidents.org file folders, I can see activity as early as Dec 24th to this site – logged by a company whose business is intrusion detection - and the activity to this site is ONLY ports 8080 and 3128.

The trace I found follows the above information similarly. However, I only had a few direct hits from this address. It does look like I could have been a small part of a larger scan if the attacker was hitting a large range of addresses. Whatever the attacker was looking for, he most likely was scanning a subnet.

Here is some information of how RingZero spreads and infects hosts from Symantec's website (http://securityresponse.symantec.com/avcenter/venc/data/ringzero.trojan.html):

There are three versions of the Trojan horse:

- **Its.exe**
  Its.exe copies itself to the \Windows\System folder when executed for the first time. It also drops the Ring0.vxd file into the same folder. Its.exe is executed again the next time that Windows starts. At this time, it creates another file to hold its data, Its.dat. It then tries to connect to two Web sites that contain strings that attempt to send mail to an address at a pager service using the Microsoft mail server.
- **Pst.exe**
  Pst.exe installs itself in the same manner as Its.exe, and also drops the Ring0.vxd file into the same folder. It attempts to connect to a different Web site than those that Its.exe tries to access.

- **Telnet23.exe**
  Telnet23.exe is another version that appears to steal Windows cached passwords. It attempts to reach a Web site and send email.

These Trojans can be packed within other host programs. When you run the host program, the Trojan is installed on the computer. RingZero hides its process by registering itself as a Windows service, so it is not displayed in the Windows task manager. It also hides its entry in the Windows registry. If the Trojan is not running, the startup call in the registry \Run key is visible.

Unfortunately, I did not have a detailed enough trace to nail this down to a specific attack though it most likely is a regular proxy scan.

7. Evidence of active targeting:

The attacker is possibly looking for responses to find proxy servers for future attacks or exploit. In this scenario, the ports were not open, nor did I return any packets to his stimulus. My host was targeted in the initial attempt to discover hosts. Since I was not vulnerable and did not respond in general, I do not expect to be actively targeted. Also, I have not seen any more attacks of this nature since.

8. Severity:

(Critical + Lethal) – (System Countermeasures + Network Countermeasures) = Severity

Criticality: 3 – Home computer was attacked directly on Internet.
Lethality: 3 – Proxy server scan or RingZero probe.
System Countermeasures: 5 – NT System with latest patches and virus protection
Network Countermeasures: 5 – Host firewall (BlackICE)

Severity: (3 + 3) – (5 + 5) = -4

9. Defensive recommendation:

As with most scans, the less information you give the attacker the better. Especially with the Internet, a good host firewall and patched operating system is needed to maintain security at all times. By default, you should block these ports from the Internet and drop all probes to these ports. To protect from a possible RingZero infection, I would use good virus protection for the file system and for email.

10. Multiple choice test question:

What is this trace most likely?

A. A scan for RingZero infected clients.
B. More information is needed to completely determine. (**CORRECT**)
C. A network scan for proxy servers.
D. A network map attempt disguised behind a proxy probe.

Detect 5

```
17:02:56.909754 x.x.x.20.1467 > x.x.x.213.5632:  [udp sum ok] udp 2 (ttl 127, id 44210, len 30)
20:35:40.933753 x.x.x.5.2359 > x.x.x.213.5632:  [udp sum ok] udp 2 (ttl 254, id 53867, len 30)
20:35:40.933753 x.x.x.5.2359 > x.x.x.213.22:  [udp sum ok] udp 2 (ttl 254, id 54123, len 30)
20:38:27.382753 x.x.x.5.2365 > x.x.x.213.22:  [udp sum ok] udp 2 (ttl 254, id 43382, len 30)
20:39:33.958753 x.x.x.5.2377 > x.x.x.213.5632:  [udp sum ok] udp 2 (ttl 254, id 54652, len 30)
20:39:33.968754 x.x.x.5.2377 > x.x.x.213.22:  [udp sum ok] udp 2 (ttl 254, id 54908, len 30)
20:44:40.238754 x.x.x.212.1076 > x.x.x.213.22:  [udp sum ok] udp 2 (ttl 127, id 11538, len 30)
21:27:40.028753 x.x.x.141.1224 > x.x.x.213.5632:  [udp sum ok] udp 2 (ttl 127, id 25580, len 30)
21:28:58.851753 x.x.x.141.1233 > x.x.x.213.5632:  [udp sum ok] udp 2 (ttl 127, id 26488, len 30)
```

1. Source of Trace:

This trace was taken from a computer at my home that is directly on the Internet through a cable modem.

2. Detect was generated by:

The detect was generated by BlackIce running on a Windows NT 4.0 server. The trace file that BlackIce provides for logging was in raw format. I read it into Ethereal for analysis, then output it again in tcpdump format. I then ran the file through windump to get the text trace of the packets shown here. The source and destination addresses have been sanitized as well because they are on the same network.

The log format is timestamp, source IP.port, destination IP.port, checksum, protocol, size, time-to-live, id field, and length.

3. Probability the source address was spoofed:

There is almost a certainty that the source address was spoofed in this trace. Even if it was not in all packets, we are not sure which one is the valid attacker, and which ones are spoofed. The random packets from multiple source addresses are a good indication. Also, being on a cable network with shared media, the ability to sniff the network could allow the attacker to spoof addresses and possibly still see the responses while he remains hidden.

4. Description of attack:

This is an attack on hosts that have the Symantec PCAnywhere remote control software installed and possibly incorrectly configured. When in remote access mode, the software listens on UDP port 5632 (UDP port 22 for older versions) for connections to the host. This attack is a scan of the local subnet for a response on UDP 5632 and 22 to see what hosts may be susceptible to a PCAnywhere attack. In this trace, the attacker sends UDP packets hoping to catch the responses to his probes without being discovered. The nature of the attack indicates this.

5. Attack mechanism:

The attacker is searching for susceptible hosts to PCAnywhere vulnerabilities. PCAnywhere has a variety of security settings. It can easily be setup quick, fast, and incorrectly. This trace shows an attacker trying to slyly identify hosts worth attacking or watching.

The disturbing part of this attack is that the attacker may not be able to do anything initially. He may just find out which hosts have PCAywhere. Due to his method of attack, this means he most likely can sniff the network. This indicates he can watch future connections and look for the PCAnywhere password in the connection. Once the attacker has this information, he can return for further exploit. He can own the machine at that point.

The spoofed packets help obscure the hacker. There are several irregularities here too. The attacker has used varying time-to-lives, id field values, and ranging source ports. It is quite hard to gather if any could be the real attacker's IP address. The assumption is that we do not have it here. Since all the packets stay on this network, it is irrelevant anyway. He should be able to see the responses. If he is not sniffing, he is trying hard to obscure his source address. The timeframe of all the packets is about four and a half hours here with most packets coming in the last hour and a half. This could be a sign of a patient and adept attack.

6. Correlations:

Here is another PCAnywhere scan from incident.org (http://www.incidents.org/diary/september2001.php#104a):

A. PC Anywhere Scan

A UDP scan of 254 targets for PC Anywhere. The source address is registered to Deutsche Telekom, Germany. Symantec's page tells more about PCAnywhere's use of port 5632/udp.

http://service1.symantec.com/SUPPORT/pca.nsf/docid/1996123152253

2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.128 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.129 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.130 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.131 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.132 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.133 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.134 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.135 - 5632 - UDP

2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.136 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.137 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.138 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.139 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.140 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.141 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.142 - 5632 - UDP
2001-09-08 14:59:02 +02:00 217.80.159.156 - 1040 - 10.147.107.143 - 5632 - UDP

The fact that this attack has multiple source addresses and takes place of most likely promiscuous media makes it very scary. More attacks of a different nature could be used to further penetrate and gather information about hosts on this network. Encryption should be used on all sensitive data.

7. Evidence of active targeting:

This trace shows the attacker looking for susceptible PCAnywhere hosts. My computer did not respond to these random packets. No information was returned to the attacker from my host. However, even though I have not seen any more local attacks, this information would make me more cautious about what I send over the Internet. Someone can always be listening.

8. Severity:

(Critical + Lethal) – (System Countermeasures + Network Countermeasures) = Severity

Criticality: 3 – Home computer was attacked directly on Internet.
Lethality: 5 – PCAnywhere probe using spoofed address with sniffing.
System Countermeasures: 5 – NT System with latest patches and virus protection
Network Countermeasures: 5 – Host firewall (BlackICE)

Severity: (3 + 5) – (5 + 5) = -2

9. Defensive recommendation:

This is a classic example of how default configurations can open holes. PCAnywhere should not be installed on an Internet facing machine without close scrutiny on how it is configured. In this case, I do not have the PCAnywhere listening for remote connections. The host firewall as well blocks in coming UDP.

On any PCAnywhere host, the configuration and security options should be reviewed according to your corporate policy. PCAnywhere connections should not be allowed from the Internet either. Corporate firewalls or a host firewall will protect the hosts from this kind of attack.

10. Multiple choice test question:

With what service is UDP port 22 most often confused?

A. FTP
B. SMTP
C. TELNET
D. SSH

Assignment 3 - "Analyze This" Scenario

The following files were used for analysis: (November 30 – December 4, 2001)

Alerts:	OOS:	Scans:
alert.011130.gz	oos_Nov.30.2001.gz	scans.011130.gz
alert.011201.gz	oos_Dec.1.2001.gz	scans.011201.gz
alert.011202.gz	oos_Dec.2.2001.gz	scans.011202.gz
alert.011203.gz	oos_Dec.3.2001.gz	scans.011203.gz
alert.011204.gz	oos_Dec.4.2001.gz	scans.011204.gz

(In most instances the MY.NET of the IP addresses was converted to 10.10 for information reporting purposes)

Overview:

The analysis data consisted of five days from November 30, 2001 to December 4, 2001. Several alerts of great interest and high occurrence are evaluated as well as relevant IP addresses from the scans as well. The following chart is a summary of the top 30 alerts found over those days.

Here is a chart of the top alerts over the five days:

Type of Alert Detected	Total
MISC Large UDP Packet	328,058
MISC source port 53 to <1024	59,745
CS WEBSERVER - external web traffic	57,559
MISC traceroute	48,627
INFO MSN IM Chat data	33,817
ICMP Echo Request BSDtype	33,707
WEB-MISC prefix-get //	32,981
Tiny Fragments - Possible Hostile Activity	29,009
Watchlist 000220 IL-ISDNNET-990517	21,191

SMB Name Wildcard	18,127
ICMP Destination Unreachable (Host Unreachable)	7,506
ICMP Echo Request Nmap or HPING2	5,955
ICMP Echo Request CyberKit 2.2 Windows	3,772
ICMP Destination Unreachable (Communication Administratively Prohibited)	3,002
INFO Napster Client Data	2,974
NMAP TCP ping!	2,788
ICMP Echo Request L3retriever Ping	2,233
SCAN Proxy attempt	1,937
Watchlist 000222 NET-NCFC	1,887
SunRPC High Port Access	1,722
Incomplete Packets Fragments Discarded	1,520
ICMP Fragment Reassemble Time Exceeded	1,437
External RPC call	1,379
ICMP Destination Unreachable (Network Unreachable)	1,348
ICMP Echo Request Sun Solaris	1,315
WEB-MISC 403 Forbidden	1,244
INFO FTP anonymous FTP	1,080
Queso fingerprint	933
INFO Inbound GNUTella Connect accept	805
ICMP Destination Unreachable (Protocol Unreachable)	754

The following alerts were all noted in high numbers.

Top Alerts (occurrence and severity):

1. MISC Large UDP Packet

The following Snort rule generated the alerts:
(`alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Large UDP Packet"; dsize: >4000; reference:arachnids,247; classtype:bad-unknown; sid:521; rev:1;)`)
There is definitely a large number of packets involved. The IP address 10.10.70.134 was a high destination address for these attacks, and it also was noted to have other signatures related to it. The source address associated with most of these attacks was 209.190.237.123. Here are more alerts for this source address:

- 1 instances of *Attempted Sun RPC high port access*
- 1 instances of *ICMP Echo Request Sun Solaris*
- 1 instances of *TFTP - Internal UDP connection to external tftp server*
- 26 instances of *High port 65535 udp - possible Red Worm - traffic*
- 32951 instances of *MISC Large UDP Packet*

This address sends random UDP packets to the 10.10.70.134 address, some with source and destination port 0. Something is going on here (Advanced UDP scan, DoS). The Red Worm alert makes me suspicious that this host may be infected.

Also, 10.10.111.121 had over 30,000 hits itself for this attack. The source of this attack was mainly 61.153.17.188 over several days. However, it had coordinating source and destination ports of a more normal nature. This type of traffic leads at a university, I would like to believe that it is most likely gaming software over the Internet. Since this attack was quite prevalent, it would be important to verify this theory.
The following alerts were observed with the UDP Packets:

12/01-11:04:46.395193 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188
12/01-11:51:37.759465 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188
12/01-12:04:59.717245 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188
12/01-12:32:58.731691 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188
12/01-13:06:59.175351 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188
12/01-14:14:41.039529 [**] ICMP Fragment Reassembly Time Exceeded [**] 10.10.111.221 -> 61.153.17.188

This could either confirm a lot of data in fragmented packets or a sophisticated fragment attack. Either way, this signature was a lot more coordinated and seem to indicate data transfer of a more normal nature than the previous IP address of the same UDP alert.

2. MISC Source port 53 to < 1024

Destinations	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
10.10.1.3	3842	3843	1903	1903
10.10.1.4	3328	3338	1505	1509
10.10.1.5	3258	3258	1486	1486
10.10.1.2	220	221	85	86
10.10.130.122	190	190	2	2
10.10.137.7	164	256	88	90
10.10.1.10	11	11	2	2
10.10.1.9	3	4	1	2

This table shows a list of the top destinations that the majority of the alerts came from over the five days. They are all internal servers that correlate to about 4500 external

source IP addresses. The source addresses are random and diverse. After looking at the traces (below), I have a feeling this is DNS traffic. It could either be large name resolution or zone transfers. It does not look like a coordinated attack but more like DNS servers configured to use the source port as 53. A lot of older DNS servers seem to still do this.

3. CS WEBSERVER – external web traffic

This is not generated by a standard Snort rule that I could find. I believe that it is a custom rule inserted to watch web traffic to a CS (Computer Science) web server. The server 10.10.100.165 is the destination of all 57,559 alerts of this kind. On further investigation, this server seems to be vulnerable to at least two attacks as well. The list of attacks below are a sample of those executed against the server.

19 different signatures are present for *10.10.100.165* as a destination

- 1 instances of *CS WEBSERVER - external cmd traffic*
- 1 instances of *WEB-MISC prefix-get //*
- 1 instances of *WEB-CGI ksh access*
- 1 instances of *ICMP IPV6 Where-Are-You*
- 1 instances of *WEB-MISC Lotus Domino directory traversal*
- 1 instances of *WEB-CGI formmail access*
- 1 instances of *INFO FTP anonymous FTP*
- 1 instances of *WEB-CGI csh access*
- 2 instances of *SUNRPC highport access!*
- 2 instances of *Null scan!*
- 2 instances of *Probable NMAP fingerprint attempt*
- 3 instances of *NMAP TCP ping!*
- 5 instances of *WEB-IIS _vti_inf access*
- 9 instances of *WEB-FRONTPAGE _vti_rpc access*
- 10 instances of *WEB-CGI redirect access*
- 11 instances of *WEB-IIS view source via translate header*
- 31 instances of *WEB-MISC http directory traversal*
- 37 instances of *CS WEBSERVER - external ftp traffic*
- 8450 instances of *CS WEBSERVER - external web traffic*

This server is either a honey pot for attackers, or it has been compromised. The nature of the attacks indicates that it is well hammered from the Internet and that it has vulnerabilities.

4. Tiny Fragments – Possible Hostile Activity

After reviewing the large number of alerts for this attack (29,009), I am not certain what is going on between these two servers.

12/04-12:21:07.507397 [**] Tiny Fragments - Possible Hostile Activity [**] 10.10.8.1 -> 10.10.16.42

12/04-12:21:07.802910 [**] Tiny Fragments - Possible Hostile Activity [**] 10.10.8.1 -> 10.10.16.42
12/04-12:21:08.916584 [**] Tiny Fragments - Possible Hostile Activity [**] 10.10.8.1 -> 10.10.16.42
12/04-12:21:11.467212 [**] Tiny Fragments - Possible Hostile Activity [**] 10.10.8.1 -> 10.10.16.42

This is the format for the alerts that indicate a large amount of tiny fragments between 10.10.8.1 and 10.10.16.42. It seems one directional, and there is a strong possibility that 10.10.16.42 is compromised. Other signatures indicate possible Trojan activity on this host. The stimulus on 10.10.8.1 is the confusing part. Since both addresses are internal and I bet this Snort rule is custom, there is an active monitoring of what happens between these servers. The barrage starts about noon and goes till midnight. All packets follow the above trace. More information is needed to determine the nature and severity of this attack.

Top Scans:

1. Sources 10.10.5.75 and 10.10.5.76 (Internal)

These two servers showed up consistently in the scan files as the top two portscan hosts detected. The host 10.10.5.75 had 107,309 alerts for detected portscans, which equated to 1,935,113 entries in the scan files. The other address, 10.10.5.76, had 107,282 alerts respectively. I have grouped these two together because their scan are a mimic of each other. Here is a sample trace:

```
Dec  2 12:00:01 10.10.5.75:67 -> 10.10.230.198:68 UDP
Dec  2 12:00:01 10.10.5.75:67 -> 10.10.226.250:68 UDP
Dec  2 12:00:01 10.10.5.75:67 -> 10.10.218.62:68 UDP
Dec  2 12:00:03 10.10.5.75:67 -> 10.10.223.14:68 UDP
Dec  2 12:00:03 10.10.5.75:67 -> 10.10.223.82:68 UDP
Dec  2 12:00:03 10.10.5.75:67 -> 10.10.235.226:68 UDP
Dec  2 12:00:04 10.10.5.75:67 -> 10.10.235.178:68 UDP
```

This activity looks like a robust UDP scan of port 68 on a Class B subnet. However due to the ports, a reasonable argument for DHCP responses could be made as well. There were no specific alerts that I could find related to attacks by these hosts. The rapid number of packets in a quick timeframe has them labeled as portscans. I have trouble believing that two hosts would continue scanning like this for five days straight. At some point, they would have covered every host. Since a Class B network is so large, it seems for feasible that DHCP is a likely answer.

2. Source 10.10.87.50 (Internal)

This was another internal address with high scanning numbers. There were 75,060 alerts logged for this address alone. Again, this host used all UDP packets to a lot of random addresses including the 24.x.x.x, known to host cable modem networks. Here is a sample:

```
Dec  3 06:01:53 10.10.87.50:999 -> 217.128.162.218:61098 UDP
Dec  3 06:01:54 10.10.87.50:888 -> 24.217.195.224:3561 UDP
Dec  3 06:01:54 10.10.87.50:888 -> 195.67.214.193:10514 UDP
```

```
Dec  3 06:05:20 10.10.87.50:888 -> 61.182.30.17:1217 UDP
Dec  3 06:05:20 10.10.87.50:999 -> 210.243.192.231:1042 UDP
Dec  3 06:05:20 10.10.87.50:888 -> 210.243.192.231:1045 UDP
```

In this section of the trace, the source port stays at 999 or 888 and the destination port changes randomly. It seems to be a constant UDP scan of Internet networks. The ports were so random I could not establish a pattern of the scan. The source port points to a crafted scan since it never changes. This scan again was continuing over all five days.

3. Source 205.188.246.121 (External)

```
Nov 30 10:41:51 205.188.246.121:28556 -> 10.10.108.15:6970 UDP
Nov 30 10:41:51 205.188.246.121:17296 -> 10.10.156.54:6970 UDP
Nov 30 10:41:55 205.188.246.121:31166 -> 10.10.86.28:6970 UDP
Nov 30 10:41:54 205.188.246.121:17296 -> 10.10.156.54:6970 UDP
Nov 30 10:41:54 205.188.246.121:12354 -> 10.10.83.72:6970 UDP
Nov 30 10:41:55 205.188.246.121:27758 -> 10.10.181.76:6970 UDP
```

When I first looked into the port 6970, I found it was used by RealNetworks and RTSP. This would explain the large number of packets. This host scanned over 6000 times just on the first day (11/30), and there were 4,556 alerts by Snort of his presence through the five days. I had also recorded over 25,000 scan hits for the whole 205.188.0.0 Class B network from the Internet. It looked like something serious was going on here. After further discovery, this seems to be a false alarm.

Two of the top addresses are shown here:

```
C:\>nslookup
Default Server:  atldns03.atl.mediaone.net
Address:  66.56.65.7

> 205.188.246.121
Server:  atldns03.atl.mediaone.net
Address:  66.56.65.7

Name:    g2lb3.spinner.com
Address:  205.188.246.121

> 205.188.228.65
Server:  atldns03.atl.mediaone.net
Address:  66.56.65.7

Name:    mslb4.spinner.com
Address:  205.188.228.65

> exit
```

This is an nslookup showing that the IP addresses resolve to spinner.com, which happens to be an online radio station. It is most likely that the traffic here is setting

off portscan alerts due to the volume in a short amount of time. I did find that source ports matched up consistently across logged packets indicated crafting was not involved.

4. Source 204.152.184.75 (External) ftp.netbsd.org

```
Dec  4 05:10:24 204.152.184.75:59390 -> 10.10.70.148:2707 SYN ******S*
Dec  4 05:10:25 204.152.184.75:59379 -> 10.10.70.148:2709 SYN ******S*
Dec  4 05:10:26 204.152.184.75:59375 -> 10.10.70.148:2710 SYN ******S*
Dec  4 05:10:29 204.152.184.75:59348 -> 10.10.70.148:2713 SYN ******S*
Dec  4 05:10:29 204.152.184.75:59342 -> 10.10.70.148:2714 SYN ******S*
Dec  4 05:10:30 204.152.184.75:59334 -> 10.10.70.148:2715 SYN ******S*
Dec  4 05:10:32 204.152.184.75:59323 -> 10.10.70.148:2716 SYN ******S*
Dec  4 05:10:32 204.152.184.75:59316 -> 10.10.70.148:2717 SYN ******S*
Dec  4 05:10:33 204.152.184.75:59310 -> 10.10.70.148:2718 SYN ******S*
Dec  4 05:10:35 204.152.184.75:59285 -> 10.10.70.148:2721 SYN ******S*
Dec  4 05:10:36 204.152.184.75:59276 -> 10.10.70.148:2722 SYN ******S*
Dec  4 05:10:38 204.152.184.75:59262 -> 10.10.70.148:2724 SYN ******S*
```

This scan was another one across all the days, alerted 2,437 times. It seems to be a standard SYN scan incrementing the destination port while decrementing the source port. The strange thing, other than the source port, is that all packets were destined for 10.10.70.148. It looks like a consistent scan of this server from the host ftp.netbsd.org. It is a very loud scan. Any IDS would catch this as a basc TCP port scan. More research should be done to see what value holds to 10.10.70.148 that makes it so attractive to a constant TCP SYN attack.

Top Talkers: (Alerts by Destination Address)

Number of Alerts	Destination Address	Number of Signatures	Source Address
58391	10.10.100.165	19 signatures	(3453 source IPs)
51582	10.10.140.9	4 signatures	(61 source IPs)
48794	10.10.111.221	1 signatures	61.153.17.188, 61.150.5.19
33880	10.10.253.114	11 signatures	(533 source IPs)
32980	10.10.70.134	5 signatures	209.190.237.123, 193.253.224.66
29018	10.10.16.42	5 signatures	(3 source IPs)
27174	10.10.70.148	3 signatures	(24 source IPs)
20750	10.10.1.3	8 signatures	(3481 source IPs)
17655	10.10.1.5	4 signatures	(2615 source IPs)
17343	10.10.1.4	3 signatures	(2655 source IPs)

These were the most active destination addresses across the five days that were analyzed. Special precaution should be taken to ensure security of these machines.

Important External Addresses:

61.153.17.188 – Top Source Address for 11/30/2001, 12/01/2001, and 12/03/2001 (http://www.apnic.net) (Strange UDP Packets)

inetnum: 61.153.17.0 - 61.153.17.255
netname: NINGBO-ZHILAN-NET
descr: NINGBO TELECOMMUNICATION CORPORATION
,ZHILAN APPLICATION SERVICE PROVIDER
descr: Ningbo, Zhejiang Province
country: CN
admin-c: CZ61-AP
tech-c: CZ61-AP
mnt-by: MAINT-CHINANET-ZJ
changed: master@dcb.hz.zj.cn 20010512
source: APNIC

person: CHINANET ZJMASTER
address: no 378,yan an road,hangzhou,zhejiang
country: CN
phone: +86-571-7015441
fax-no: +86-571-7027816
e-mail: master@dcb.hz.zj.cn
nic-hdl: CZ61-AP
mnt-by: MAINT-CHINANET-ZJ
changed: master@dcb.hz.zj.cn 20001219
source: APNIC

216.231.14.226 – Top Source Address for 12/02/2001 (http://www.arin.net) (Strange UDP Packets)

Cox Communications, Inc. (NETBLK-COX-OC) COX-OC 216.231.0.0 - 216.231.31.255
Regenesis (NETBLK-COXOCCA-REGENESIS-1) COXOCCA-REGENESIS-1
216.231.14.224 - 216.231.14.239

209.190.237.123 – Top Source Address for 12/04/2001 (Large UDP Packets) (http://www.arin.net)

Atlantech Online, Inc. (NETBLK-AOI1999B)
1010 Wayne Avenue, Suite 630
Silver Spring, MD 20910
US

Netname: AOI1999B
Netblock: 209.190.192.0 - 209.190.255.255
Maintainer: ATON

Coordinator:
Center, Network Operations (EF105-ARIN) noc@atlantech.net
301-589-3060 (FAX) 301-593-9897

Domain System inverse mapping provided by:

DNS1.ATLANTECH.NET 209.183.205.35
DNS2.ATLANTECH.NET 209.183.192.65

ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 22-May-2000.
Database last updated on 10-Feb-2002 19:55:25 EDT.

65.9.223.145 – Executing CMD.EXE on web server (MS Vulnerability)
(http://www.arin.net)

@Home Network (NETBLK-HOME-3BLK)HOME-3BLK 65.0.0.0 - 65.15.255.255
@Home Network (NETBLK-PITBPA1-PA-7) PITBPA1-PA-7 65.9.216.0 - 65.9.223.255

61.159.31.6 – Retrieving Directory Listing from web server (MS vulnerability)
(http://www.apnic.net)

inetnum: 61.159.0.0 - 61.159.63.255
netname: CHINANET-HE
descr: CHINANET Hebei province network
descr: Data Communication Division
descr: China Telecom
country: CN

admin-c: CH93-AP
tech-c: ZC24-AP
mnt-by: MAINT-CHINANET
mnt-lower: MAINT-CHINANET-HE
changed: hostmaster@ns.chinanet.cn.net 20001123
source: APNIC

person: Chinanet Hostmaster
address: A12,Xin-Jie-Kou-Wai Street
country: CN
phone: +86-10-62370437
fax-no: +86-10-62053995
e-mail: hostmaster@ns.chinanet.cn.net
nic-hdl: CH93-AP
mnt-by: MAINT-CHINANET
changed: hostmaster@ns.chinanet.cn.net 20000101
source: APNIC

person: zhiyong chen
address: hebei province shijiazhuang
address: fanxi road No.19
address: hebei shuju tongxin ju
country: CN
phone: +86-311-6051394
fax-no: +86-311-6672895
e-mail: jixin@sj-user.he.cninfo.net
nic-hdl: ZC24-AP
mnt-by: MAINT-CHINANET-HE
changed: chenzhy@public.sj.he.cn 20010212
source: APNIC

216.26.100.34 – Sending ICMP Redirects
(http://www.arin.net)

Education Network of Ontario (NETBLK-ENO-CIDR-1) ENO-CIDR-1
216.26.96.0 - 216.26.127.255

Keewaytinook Okimakanak (NETBLK-NETBLOCK-KNET-1) NETBLOCK-KNET-1
216.26.100.0 - 216.26.107.255

(http://www.dshield.org)

IP Address: 216.26.100.34

HostName: 34-100-26-216.ipt.knet.ca

DShield Profile:

Country:	CA
Contact E-mail:	registrar@enoreo.on.ca
Total Records against IP:	
Number of targets:	
Date Range:	to

Ports Attacked (up to 10):

Port	Attacks

Whois: Education Network of Ontario (NETBLK-ENO-CIDR-1)
20 Toronto Street, Suite 400
Toronto, ON M5C 2B8
CA

Netname: ENO-CIDR-1
Netblock: 216.26.96.0 - 216.26.127.255
Maintainer: ENO

Coordinator:
Education Network of Ontario (ZE31-ARIN) registrar@enoreo.on.ca
+1 416 848 4800

Domain System inverse mapping provided by:

DNS.ENOREO.ON.CA	205.150.114.15
HUB.ENOREO.ON.CA	205.150.114.4
CANADA.ENOREO.ON.CA	205.150.114.32
OTTAWA.ENOREO.ON.CA	206.222.70.50

ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 12-Nov-2001.
Database last updated on 10-Feb-2002 19:55:25 EDT.

Keewaytinook Okimakanak (NETBLK-NETBLOCK-KNET-1)
115 King Street
Sioux Lookout, ON P8T 1A8
CA

Netname: NETBLOCK-KNET-1
Netblock: 216.26.100.0 - 216.26.107.255
Maintainer: KTNK

Coordinator:
Linden, Adi (AL539-ARIN) adi@adis.on.ca
807-737-1135

Domain System inverse mapping provided by:

PLUTO.KNET.ON.CA	216.211.97.2
GOOFY.KNET.ON.CA	216.211.97.4
GIZMO.ADIS.ON.CA	216.211.97.50

Record last updated on 06-Apr-2001.
Database last updated on 10-Feb-2002 19:55:25 EDT.

204.152.184.75 – TCP scanning host
(http://www.dshield.org)

IP Address: 204.152.184.75
HostName: ftp.netbsd.org
DShield Profile:

Country:	US
Contact E-mail:	paul@VIX.COM
Total Records against IP:	
Number of targets:	
Date Range:	to

Ports Attacked (up to 10):

Port	Attacks

Whois: M.I.B.H., LLC (NETBLK-MIBH-2BLK)
Star Route Box 159A
Woodside, CA 94062
US

Netname: MIBH-2BLK

Netblock: 204.152.184.0 - 204.152.191.255

Maintainer: VIX

Coordinator:

Vixie, Paul (PV15-ARIN) paul@VIX.COM

+1 415 747 0204

Domain System inverse mapping provided by:

NS-EXT.VIX.COM 204.152.184.64

NS1.GNAC.COM 209.182.195.77

Record last updated on 27-Apr-1999.

Database last updated on 10-Feb-2002 19:55:25 EDT.

OOS Data:

After catenation of the OOS files into one file, I began to sort the data and arrange it in an organized fashion. I was able to determine that a large majority of the traffic was external addresses sending TCP packets to port 25 with a SYN flag and the two reserved bits set. Here is part of the breakdown:

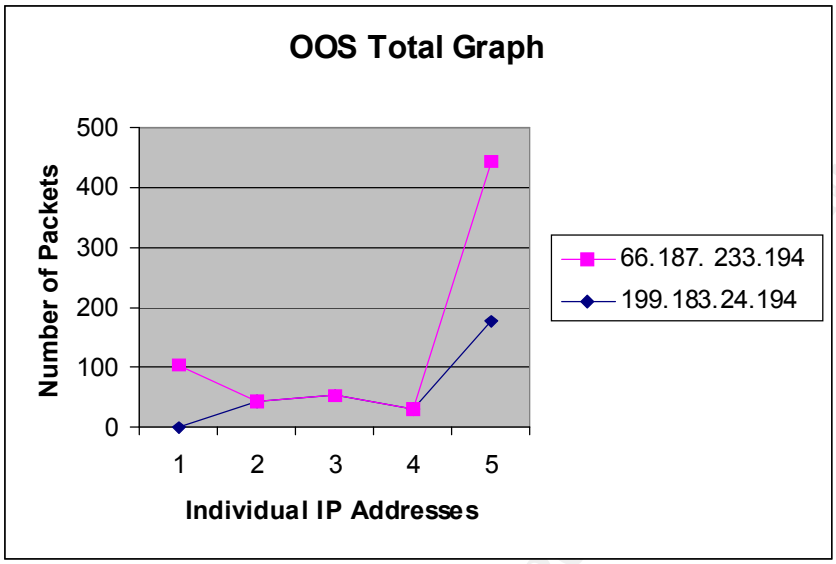
66.187. 233.194 → 10.10.253.43 port 25 1 2 S (103 packets)
66.187. 233.194 → 10.10.100.217 port 25 1 2 S (268 packets)

199.183.24.194 → 10.10.254.41 port 25 1 2 S (43 packets)
199.183.24.194 → 10.10.254.42 port 25 1 2 S (51 packets)
199.183.24.194 → 10.10.254.43 port 25 1 2 S (29 packets)
199.183.24.194 → 10.10.100.217 port 25 1 2 S (176 packets)

It looks like these packets may be from the Queso tool. Since the Type-Of-Service on these packets was 0, they are most likely not Explicit Congestion Notification (ECN), but rather the forged packets are used to fingerprint an operating system by a tool named queso. (http://www.sans.org/y2k/ecn.htm) The interesting factor here is the number of packets over the whole five days. Queso only needs a few packets to identify an OS. In this case, there might be a use of ECN since the packet count is higher than an OS fingerprint and the source addresses stay the same. It is hard to draw a conclusion due to my unfamiliarity with ECN. More research on these packets would be appropriate. The nature of the attack looks like there might be decent explanation.

The other OOS packets were a few random packets that might have been Queso examples as well. Also, a couple random packets with strange flags usually for OS fingerprinting like XMAS and SPU.

Link Graph:



This graph is a stacked line graph showing the combination of packets from the two source addresses using the reserved bits from the OOS files. The two addresses here are graphed to prove that Queso is most likely not being used due to the amount of data being transferred. The 66.187.233.194 is graphed as the sum of its values with the previous line to see the total packets sent.

Internal Machines:

There are several things to look out for on the internal side. The host 10.10.16.42 has several alerts that are interesting:

5 different signatures are present for *10.10.16.42* as a destination

- 1 instances of *Tiny Fragments - Possible Hostile Activity [**] 10.10.8.112/04-19:44:32.070566 [**] INFO MSN IM Chat data*
- 2 instances of *Possible trojan server activity*
- 3 instances of *Port 55850 tcp - Possible myserver activity - ref. 010313-1*
- 3 instances of *SUNRPC highport access!*
- 29009 instances of *Tiny Fragments - Possible Hostile Activity*

The 29,009 Tiny fragments along with the Trojan activity make this an interesting host. The other host, 10.10.8.1, that is sending it the Tiny Fragments should be noted as well. This is the only signature that was detected for it, but the nature of the attack is very suspicious as well. More information about this attack is needed.

The internal host 10.10.100.165 is another dangerous one.

4 different signatures are present for *10.10.100.165* as a source

- 1 instances of *INFO - Web Dir listing*
- 3 instances of *High port 65535 tcp - possible Red Worm - traffic*
- 12 instances of *INFO - Web Cmd completed*
- 57 instances of *WEB-MISC 403 Forbidden*

There alerts explain that there are vulnerabilities on this server which have not been addressed. According to this, it may have already been compromised.

The host 10.10.130.27 may have an active Trojan. The trace shows a decent chance that SubSeven is running on the host. This needs to be investigated.

12/03-01:39:16.835949 [**] IDS50/trojan_trojan-active-subseven [**] 10.10.130.27:1243 -> 24.3.40.141:1191

There is something strange about 10.10.140.9 as well. This host routinely gets way too many ICMP destination unreachables and miscellaneous traceroutes. It is a little too random and high than it should be. Something may be going on that we are not seeing. There has to be an explanation for this traffic.

Another one from before, 10.10.70.134 displays UDP traffic of an unnatural nature. Random source and destination ports as well as using 0 as a port is not valid traffic. This should be probed further for cause and effect.

12/04-12:27:18.791076 [**] MISC Large UDP Packet [**] 209.190.237.123:0 -> 10.10.70.134:0
12/04-12:27:18.901994 [**] MISC Large UDP Packet [**] 209.190.237.123:39356 -> 10.10.70.134:22150
12/04-12:27:19.757255 [**] MISC Large UDP Packet [**] 209.190.237.123:37885 -> 10.10.70.134:3313
12/04-12:27:20.339481 [**] MISC Large UDP Packet [**] 209.190.237.123:0 -> 10.10.70.134:0
12/04-12:27:20.451450 [**] MISC Large UDP Packet [**] 209.190.237.123:0 -> 10.10.70.134:0
12/04-12:27:21.430906 [**] MISC Large UDP Packet [**] 209.190.237.123:0 -> 10.10.70.134:0
12/04-12:27:21.740402 [**] MISC Large UDP Packet [**] 209.190.237.123:34652 -> 10.10.70.134:11460

Defensive Recommendations:

I first recommend better segmentation of the network and adding layers of defense to improve things immediately. Applying ingress and egress filtering at the perimeter routers will help filter out a lot of miscellaneous traffic. It would help prevent DoS attacks or IP spoofing.

It was hard for me to gather how the network was segmented, but I would recommend layering firewalls and separating segments to help contain like computers to their respective areas. Also, putting a firewall and host protection on anything that servers information to the Internet. Specifically, there was at least one server that was not patched against some common vulnerabilities. That server needs to be updated immediately.

Port filtering on common Trojan ports, unused services, and unsecured protocols would b the next step. A more strict Internet policy would need to be developed and approved. I would recommend limiting UDP as much as possible. A lot of the ICMP could be removed as well. A good stateful firewall will help with a lot of issues.

I would also block some of the dangerous IP addresses identified immediately. The firewall ability to protect your network after detecting these attacks is key. The IDS will need to be tuned with some custom rules to make sure the firewall is working. Quality Assurance is a good aspect of security that a decent IDS can provide for your network. It will help verify what does or does not get through the firewall.

Analysis Process:

The analysis of the data took some discipline and research. I read several practicals to get an idea of what I had to do and some of the ways to get it done. In the end, I found a mixed approach worked the best.

I downloaded SnortSnarf v020126.1 and installed it on my laptop. I was able to run in on the daily alert files, but it seemed to take too long on one big concatenated file. I ended up using Excel to correlate and combine the alerts into a table.

To parse some of the files and pull out specific information, especially in the scans and OOS files, I knew I would need certain unix commands like cat, grep, sed, sort, uniq. I had some of them from the Windows 2000 Resource Kit, but I had trouble locating sed. In the end, I installed Cygwin 1.3.5 from a download I had on my hard drive. This allowed me to use almost any posix utility I needed to parse the files. I ended up grepping out the spp_portscans for easier correlation. Sed was a great help to me for preparing the files for SnortSnarf by removing the MY.NET. I was able to substitute "10.10" for each "MY.NET".

SnortSnarf proved to be invaluable, and most of the charts and correlations came from its data. What a wonderful tool. I did try one more time to complete the huge file on my home machine, but it never completed before I was finished with this.

I used the Internet to research heavily and correlate my findings. Incident.org, sans.org, and giac.org were very helpful in my data mining.

References:

Websites referenced in this practical: (urls where appropriate)

1. General Search and Archives. http://www.incidents.org/ (8 Feb. 2002)

2. Virus Database. http://www.symantec.com/ (8 Feb. 2002)

3. Virus Database and Research. http://www.sarc.com/ (8 Feb. 2002)

4. SANS Reading Room. http://www.sans.com/ (8 Feb. 2002)

5. ARIN WHOIS Lookup. http://arin.net/ (8 Feb. 2002)

6. APNIC WHOIS Lookup. http://www.apnic.net/ (8 Feb. 2002)

7. Alerts and Detects of Interest. http://xforce.iss.net/ (7 Feb. 2002)

8. WHOIS Data. http://www.dshield.org/ (10 Feb. 2002)

9. GCIA Information and Practicals. http://www.giac.org/ (10 Feb. 2002)

Several GCIA practicals: (www.giac.org)

Nelson Carter (0374)
Tadaaki Nagao (0187)
Doug Harold (0381)
Wade Dauphine (0387)
Win Miller (0386)
Chris Baker (0371)
Jeff Holland (0396)
Scott Shinberg (0389)

Northcutt, Stephen; Cooper, Mark; Fearnow, Matt; Frederick, Karen. Intrusion Signatures and Analysis. Indianapolis:New Riders, January 2001.

NorthCutt, Stephen and Novak, Judy. Network Intrusion Detection Handbook An Analyst's Handbook. Second Edition; Indianapolis:New Riders, September 2000.

Stevens, Richard W. TCP/IP Illustrated, Volume 1 The Protocols. Reading: Addison-Wesley, 1994.

SANS 2001 San Diego Conference Books – Intrusion Detection In-Depth Track.

Appendix A -- SnortSnarf Data Sample Screens

Earliest alert at **00:00:07**.151664 *on 11/30/2001*
Latest alert at **23:52:22**.632967 *on 11/30/2001*

Top 20 source IPs
Top 20 destination IPs

Priority	Signature (click for sig info)	# Alerts	# Sources	# Dests	Detail link
N/A	EXPLOIT x86 setgid 0	1	1	1	Summary
N/A	HelpDesk 10.10.70.49 to External FTP	1	1	1	Summary
N/A	Incomplete Packet Fragments Discarded	1	1	1	Summary
N/A	RPC portmap request rstatd	1	1	1	Summary
N/A	Virus - SnowWhite Trojan Incoming	1	1	1	Summary
N/A	WEB-CGI formmail access	1	1	1	Summary
N/A	WEB-MISC Invalid URL	1	1	1	Summary
N/A	HelpDesk 10.10.70.50 to External FTP	1	1	1	Summary
N/A	Security 000516-1	1	1	1	Summary
N/A	INFO - Web Dir listing	1	1	1	Summary
N/A	WEB-CGI finger access	1	1	1	Summary
N/A	EXPLOIT x86 stealth noop	1	1	1	Summary
N/A	WEB-CGI rsh access	1	1	1	Summary
N/A	WEB-IIS Unauthorized IP Access Attempt	1	1	1	Summary
N/A	CS WEBSERVER - external ssh traffic	1	1	1	Summary
N/A	SMTP chameleon overflow	1	1	1	Summary
N/A	WEB-CGI csh access	1	1	1	Summary

N/A	EXPLOIT x86 setuid 0	2	2	2	Summary
N/A	WEB-MISC Lotus Domino directory traversal	2	1	1	Summary
N/A	WEB-CGI survey.cgi access	2	1	1	Summary
N/A	DDOS mstream handler to client	2	1	1	Summary
N/A	MISC PCAnywhere Startup	2	2	2	Summary
N/A	IDS50/trojan_trojan-active-subseven [arachNIDS]	3	2	2	Summary
N/A	WEB-MISC guestbook.cgi access	3	2	1	Summary
N/A	WEB-FRONTPAGE fpcount.exe access	3	2	2	Summary
N/A	Virus - Possible MyRomeo Worm	4	3	4	Summary
N/A	INFO - Possible Squid Scan	4	3	4	Summary
N/A	TCP SMTP Source Port traffic	4	3	3	Summary
N/A	SMTP relaying denied	4	4	3	Summary
N/A	SYN-FIN scan!	5	1	1	Summary
N/A	TELNET access	6	1	5	Summary
N/A	SCAN FIN	7	4	5	Summary
N/A	ICMP Echo Request Sun Solaris	8	3	2	Summary
N/A	ICMP Destination Unreachable (Network Unreachable)	9	3	3	Summary
N/A	Virus - Possible pif Worm	11	4	5	Summary
N/A	EXPLOIT x86 NOOP	11	6	6	Summary
N/A	MISC Large ICMP Packet	11	4	7	Summary
N/A	INFO - Web Cmd completed	12	1	3	Summary
N/A	SCAN Synscan Portscan ID 19104	13	13	8	Summary
N/A	Virus - Possible scr Worm	14	5	7	Summary
N/A	SUNRPC highport access!	15	5	5	Summary

N/A	connect to 515 from inside	16	1	1	Summary
N/A	TCP SRC and DST outside network	16	8	14	Summary
N/A	WEB-IIS _vti_inf access	19	12	6	Summary
N/A	WEB-MISC compaq nsight directory traversal	20	6	6	Summary
N/A	INFO Napster Client Data	21	8	10	Summary
N/A	WEB-CGI redirect access	23	14	5	Summary
N/A	ICMP Echo Request Broadscan Smurf Scanner	24	2	20	Summary
N/A	ICMP Destination Unreachable (Fragmentation Needed and DF bit was set)	25	22	3	Summary
N/A	WEB-IIS view source via translate header	33	6	5	Summary
N/A	WEB-FRONTPAGE _vti_rpc access	33	22	8	Summary
N/A	beetle.ucs	34	2	2	Summary
N/A	High port 65535 udp - possible Red Worm - traffic	35	6	6	Summary
N/A	CS WEBSERVER - external ftp traffic	39	16	1	Summary
N/A	WEB-MISC count.cgi access	41	21	2	Summary
N/A	BACKDOOR NetMetro Incoming Traffic	42	3	3	Summary
N/A	Port 55850 udp - Possible myserver activity - ref. 010313-1	51	3	4	Summary
N/A	spp_http_decode: IIS Unicode attack detected	52	21	15	Summary
N/A	X11 outgoing	54	5	6	Summary
N/A	INFO Possible IRC Access	67	23	33	Summary
N/A	TELNET login incorrect	76	7	55	Summary
N/A	WEB-CGI scriptalias access	79	2	2	Summary

N/A	ICMP Source Quench	85	32	4	Summary
N/A	Port 55850 tcp - Possible myserver activity - ref. 010313-1	85	19	21	Summary
N/A	connect to 515 from outside	94	1	88	Summary
N/A	ICMP Fragment Reassembly Time Exceeded	95	9	11	Summary
N/A	FTP DoS ftpd globbing	95	1	1	Summary
N/A	High port 65535 tcp - possible Red Worm - traffic	96	13	13	Summary
N/A	ICMP Destination Unreachable (Protocol Unreachable)	96	5	5	Summary
N/A	WEB-MISC Attempt to execute cmd	97	16	11	Summary
N/A	WEB-MISC http directory traversal	112	27	4	Summary
N/A	ICMP Echo Request Windows	119	41	31	Summary
N/A	Possible trojan server activity	121	8	103	Summary
N/A	Null scan!	129	32	16	Summary
N/A	INFO Inbound GNUTella Connect accept	133	12	121	Summary
N/A	INFO Outbound GNUTella Connect accept	147	131	12	Summary
N/A	ICMP traceroute	169	81	101	Summary
N/A	INFO FTP anonymous FTP	204	66	69	Summary
N/A	WEB-MISC 403 Forbidden	214	6	154	Summary
N/A	spp_http_decode: CGI Null Byte attack detected	240	4	4	Summary
N/A	Queso fingerprint	253	15	20	Summary
N/A	SCAN Proxy attempt	262	18	18	Summary
N/A	NMAP TCP ping!	334	12	143	Summary
N/A	ICMP Destination Unreachable (Communication Administratively	422	55	31	Summary

	Prohibited)				
N/A	Watchlist 000222 NET-NCFC	456	13	11	Summary
N/A	External RPC call	684	2	601	Summary
N/A	ICMP Echo Request L3retriever Ping	1239	4	5	Summary
N/A	ICMP Destination Unreachable (Host Unreachable)	1471	141	31	Summary
N/A	ICMP Echo Request Nmap or HPING2	1522	16	167	Summary
N/A	Watchlist 000220 IL-ISDNNET-990517	2214	15	7	Summary
N/A	ICMP Echo Request CyberKit 2.2 Windows	3474	15	5	Summary
N/A	SMB Name Wildcard	4268	103	1439	Summary
N/A	MISC Large UDP Packet	4814	8	6	Summary
N/A	WEB-MISC prefix-get //	6022	434	2	Summary
N/A	ICMP Echo Request BSDtype	6210	12	8	Summary
N/A	INFO MSN IM Chat data	7033	161	194	Summary
N/A	MISC traceroute	9074	73	11	Summary
N/A	MISC source port 53 to <1024	11016	4058	8	Summary
N/A	CS WEBSERVER - external web traffic	12777	2035	1	Summary

Earliest alert at **00:00:07**.209120 *on 12/01/2001*
Latest alert at **23:51:54**.671592 *on 12/01/2001*

Top 20 source IPs
Top 20 destination IPs

Priority	Signature (click for sig info)	# Alerts	# Sources	# Dests	Detail link
N/A	MISC Cisco Catalyst Remote Access	1	1	1	Summary

N/A	WEB-FRONTPAGE form_results access	1	1	1	Summary
N/A	SCAN - wayboard request - allows reading of arbitrary files as http service	1	1	1	Summary
N/A	Virus - Possible pif Worm	1	1	1	Summary
N/A	EXPLOIT NTPDX buffer overflow	1	1	1	Summary
N/A	INFO - Web Dir listing	1	1	1	Summary
N/A	External FTP to HelpDesk 10.10.70.50	1	1	1	Summary
N/A	WEB-MISC webdav search access	1	1	1	Summary
N/A	INFO - Web Cmd completed	1	1	1	Summary
N/A	IDS50/trojan_trojan-active-subseven [arachNIDS]	1	1	1	Summary
N/A	EXPLOIT FTP passwd retrieval retr path	1	1	1	Summary
N/A	RFB - Possible WinVNC - 010708-1	1	1	1	Summary
N/A	ICMP IPV6 Where-Are-You	1	1	1	Summary
N/A	WEB-CGI ksh access	1	1	1	Summary
N/A	connect to 515 from outside	1	1	1	Summary
N/A	FTP passwd attempt	1	1	1	Summary
N/A	WEB-CGI rsh access	1	1	1	Summary
N/A	RPC tcp traffic contains bin_sh	1	1	1	Summary
N/A	High port 65535 udp - possible Red Worm - traffic	1	1	1	Summary
N/A	EXPLOIT x86 stealth noop	1	1	1	Summary
N/A	WEB-MISC Lotus Domino directory traversal	1	1	1	Summary
N/A	INFO Inbound GNUTella Connect request	2	2	2	Summary
N/A	DNS zone transfer	2	1	1	Summary

N/A	SMTP chameleon overflow	2	2	2	Summary
N/A	WEB-CGI glimpse access	2	2	1	Summary
N/A	EXPLOIT x86 setgid 0	2	2	2	Summary
N/A	x86 NOOP - unicode BUFFER OVERFLOW ATTACK	2	1	1	Summary
N/A	External FTP to HelpDesk 10.10.53.29	2	1	1	Summary
N/A	Virus - Possible NAIL Worm	3	2	2	Summary
N/A	SMTP relaying denied	3	3	3	Summary
N/A	WEB-MISC compaq nsight directory traversal	3	2	2	Summary
N/A	MISC PCAnywhere Startup	3	2	1	Summary
N/A	INFO - Possible Squid Scan	3	3	3	Summary
N/A	Port 55850 udp - Possible myserver activity - ref. 010313-1	4	1	1	Summary
N/A	FTP DoS ftpd globbing	4	1	1	Summary
N/A	TELNET access	4	1	4	Summary
N/A	Virus - Possible scr Worm	5	2	2	Summary
N/A	SCAN Synscan Portscan ID 19104	5	5	4	Summary
N/A	EXPLOIT x86 setuid 0	6	4	4	Summary
N/A	WEB-CGI redirect access	6	5	5	Summary
N/A	EXPLOIT x86 NOOP	6	4	4	Summary
N/A	beetle.ucs	7	1	1	Summary
N/A	ICMP Echo Request Broadscan Smurf Scanner	7	1	7	Summary
N/A	WEB-CGI csh access	7	4	2	Summary
N/A	MISC Large ICMP Packet	7	5	3	Summary
N/A	SUNRPC highport access!	7	3	3	Summary
N/A	WEB-IIS view source via translate	14	5	4	Summary

	header				
N/A	connect to 515 from inside	20	1	1	Summary
N/A	INFO Napster Client Data	21	11	17	Summary
N/A	WEB-IIS _vti_inf access	21	12	6	Summary
N/A	WEB-FRONTPAGE _vti_rpc access	22	11	7	Summary
N/A	Port 55850 tcp - Possible myserver activity - ref. 010313-1	22	10	10	Summary
N/A	ICMP Destination Unreachable (Fragmentation Needed and DF bit was set)	23	14	1	Summary
N/A	CS WEBSERVER - external ftp traffic	26	12	1	Summary
N/A	WEB-MISC count.cgi access	31	14	2	Summary
N/A	TCP SRC and DST outside network	37	10	23	Summary
N/A	INFO Possible IRC Access	38	19	19	Summary
N/A	X11 outgoing	39	3	2	Summary
N/A	ICMP Echo Request CyberKit 2.2 Windows	40	17	3	Summary
N/A	INFO Outbound GNUTella Connect accept	51	47	11	Summary
N/A	Null scan!	53	20	11	Summary
N/A	spp_http_decode: IIS Unicode attack detected	53	20	15	Summary
N/A	WEB-MISC http directory traversal	54	21	3	Summary
N/A	TELNET login incorrect	57	7	45	Summary
N/A	ICMP Source Quench	58	19	3	Summary
N/A	High port 65535 tcp - possible Red Worm - traffic	62	11	11	Summary
N/A	WEB-MISC Attempt to execute cmd	68	9	8	Summary
N/A	ICMP Fragment Reassembly Time Exceeded	86	5	7	Summary

N/A	ICMP Echo Request L3retriever Ping	107	3	3	Summary
N/A	ICMP traceroute	149	55	93	Summary
N/A	Queso fingerprint	155	24	17	Summary
N/A	spp_http_decode: CGI Null Byte attack detected	158	3	3	Summary
N/A	WEB-MISC 403 Forbidden	179	6	116	Summary
N/A	TFTP - Internal TCP connection to external tftp server	190	2	2	Summary
N/A	INFO FTP anonymous FTP	191	59	54	Summary
N/A	INFO Inbound GNUTella Connect accept	194	9	179	Summary
N/A	ICMP Echo Request Windows	200	49	28	Summary
N/A	ICMP Destination Unreachable (Protocol Unreachable)	220	9	8	Summary
N/A	SCAN Proxy attempt	247	26	20	Summary
N/A	ICMP Destination Unreachable (Network Unreachable)	348	4	5	Summary
N/A	ICMP Echo Request Sun Solaris	407	1	407	Summary
N/A	ICMP Destination Unreachable (Communication Administratively Prohibited)	446	44	27	Summary
N/A	Watchlist 000222 NET-NCFC	505	6	6	Summary
N/A	ICMP Echo Request Nmap or HPING2	611	9	98	Summary
N/A	NMAP TCP ping!	646	5	230	Summary
N/A	Watchlist 000220 IL-ISDNNET-990517	828	11	7	Summary
N/A	ICMP Destination Unreachable (Host Unreachable)	1535	144	16	Summary
N/A	SMB Name Wildcard	3015	83	1235	Summary
N/A	WEB-MISC prefix-get //	4022	307	2	Summary

N/A	ICMP Echo Request BSDtype	5939	11	7	Summary
N/A	INFO MSN IM Chat data	6089	122	164	Summary
N/A	MISC source port 53 to <1024	8177	2922	9	Summary
N/A	MISC traceroute	9377	65	4	Summary
N/A	CS WEBSERVER - external web traffic	10359	1625	1	Summary
N/A	MISC Large UDP Packet	19557	1	2	Summary

Last Updated: October 22nd, 2017

Upcoming Training

SANS Berlin 2017	Berlin, Germany	Oct 23, 2017 - Oct 28, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
San Diego Fall 2017 - SEC503: Intrusion Detection In-Depth	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
Community SANS Pensacola SEC503	Pensacola, FL	Nov 27, 2017 - Dec 02, 2017	Community SANS
SIEM & Tactical Analytics Summit & Training	Scottsdale, AZ	Nov 28, 2017 - Dec 05, 2017	Live Event
SANS Munich December 2017	Munich, Germany	Dec 04, 2017 - Dec 09, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
Las Vegas 2018 - SEC503: Intrusion Detection In-Depth	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	vLive
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
SANS London February 2018	London, United Kingdom	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Dallas 2018	Dallas, TX	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VA	Mar 17, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, Australia	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS 2018	Orlando, FL	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MD	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS vLive - SEC503: Intrusion Detection In-Depth	SEC503 - 201805,	May 02, 2018 - Jun 07, 2018	vLive
SANS Security West 2018	San Diego, CA	May 11, 2018 - May 18, 2018	Live Event
Community SANS Columbia SEC503	Columbia, MD	Aug 13, 2018 - Aug 18, 2018	Community SANS
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced