Global Information Assurance Certification Paper

GIAC Intrusion Detection
Practical Assignment for SNAP San Jose 2000

Submitted by Randall Heck

Detect 1

```
Aug 25 16:58:20 our.web.server in.telnetd[3116]: refused connect from toychest.telisphere.com
Aug 25 16:58:20 our.web.server in.telnetd[3117]: refused connect from toychest.telisphere.com
Aug 25 16:58:21 our.web.server in.telnetd[3118]: refused connect from toychest.telisphere.com
Aug 25 16:58:21 our.web.server in.telnetd[3119]: refused connect from toychest.telisphere.com
Aug 25 16:58:20 our.web.server in.telnetd[3122]: refused connect from toychest.telisphere.com
Aug 25 16:58:21 our.web.server in.telnetd[3123]: refused connect from toychest.telisphere.com
Aug 25 16:58:21 our.web.server in.telnetd[3120]: refused connect from toychest.telisphere.com
Aug 25 16:58:21 our.web.server in.telnetd[3121]: refused connect from toychest.telisphere.com
Aug 25 16:58:22 our.web.server in.telnetd[3127]: refused connect from toychest.telisphere.com
Aug 25 16:58:23 our.web.server in.telnetd[3124]: refused connect from toychest.telisphere.com
Aug 25 16:58:23 our.web.server in.telnetd[3132]: refused connect from toychest.telisphere.com
Aug 25 16:58:23 our.web.server in.telnetd[3128]: refused connect from toychest.telisphere.com
Aug 25 16:58:23 our.web.server in.telnetd[3126]: refused connect from toychest.telisphere.com
Aug 25 16:58:23 our.web.server in.telnetd[3131]: refused connect from toychest.telisphere.com
Aug 25 16:58:24 our.web.server in.telnetd[3125]: refused connect from toychest.telisphere.com
Aug 25 16:58:24 our.web.server in.telnetd[3130]: refused connect from toychest.telisphere.com
Aug 25 16:58:24 our.web.server in.telnetd[3135]: refused connect from toychest.telisphere.com
Aug 25 16:58:24 our.web.server in.telnetd[3129]: refused connect from toychest.telisphere.com
Aug 25 16:58:25 our.web.server in.telnetd[3139]: refused connect from toychest.telisphere.com
Aug 25 16:58:25 our.web.server in.telnetd[3134]: refused connect from toychest.telisphere.com
Aug 25 16:58:25 our.web.server in.telnetd[3133]: refused connect from toychest.telisphere.com
Aug 25 16:58:25 our.web.server in.telnetd[3136]: refused connect from toychest.telisphere.com
Aug 25 16:58:26 our.web.server in.telnetd[3143]: refused connect from toychest.telisphere.com
Aug 25 16:58:26 our.web.server in.telnetd[3140]: refused connect from toychest.telisphere.com
Aug 25 16:58:26 our.web.server in.telnetd[3138]: refused connect from toychest.telisphere.com
Aug 25 16:58:26 our.web.server in.telnetd[3142]: refused connect from toychest.telisphere.com
Aug 25 16:58:27 our.web.server in.telnetd[3137]: refused connect from toychest.telisphere.com
Aug 25 16:58:27 our.web.server in.telnetd[3146]: refused connect from toychest.telisphere.com
Aug 25 16:58:27 our.web.server in.telnetd[3141]: refused connect from toychest.telisphere.com
Aug 25 16:58:28 our.web.server in.telnetd[3149]: refused connect from toychest.telisphere.com
Aug 25 16:58:28 our.web.server in.telnetd[3147]: refused connect from toychest.telisphere.com
Aug 25 16:58:28 our.web.server in.telnetd[3144]: refused connect from toychest.telisphere.com
Aug 25 16:58:29 our.web.server in.telnetd[3150]: refused connect from toychest.telisphere.com
Aug 25 16:58:28 our.web.server in.telnetd[3145]: refused connect from toychest.telisphere.com
Aug 25 16:58:29 our.web.server in.telnetd[3154]: refused connect from toychest.telisphere.com
Aug 25 16:58:29 our.web.server in.telnetd[3148]: refused connect from toychest.telisphere.com
Aug 25 16:58:30 our.web.server in.telnetd[3153]: refused connect from toychest.telisphere.com
Aug 25 16:58:30 our.web.server in.telnetd[3157]: refused connect from toychest.telisphere.com
Aug 25 16:58:30 our.web.server in.telnetd[3160]: refused connect from toychest.telisphere.com
Aug 25 16:58:30 our.web.server in.telnetd[3156]: refused connect from toychest.telisphere.com
Aug 25 16:58:30 our.web.server in.telnetd[3152]: refused connect from toychest.telisphere.com
Aug 25 16:58:31 our.web.server in.telnetd[3151]: refused connect from toychest.telisphere.com
Aug 25 16:58:31 our.web.server in.telnetd[3159]: refused connect from toychest.telisphere.com
Aug 25 16:58:31 our.web.server in.telnetd[3155]: refused connect from toychest.telisphere.com
Aug 25 16:58:32 our.web.server in.telnetd[3165]: refused connect from toychest.telisphere.com
Aug 25 16:58:32 our.web.server in.telnetd[3164]: refused connect from toychest.telisphere.com
Aug 25 16:58:32 our.web.server in.telnetd[3158]: refused connect from toychest.telisphere.com
Aug 25 16:58:32 our.web.server in.telnetd[3161]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3167]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3163]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3168]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3166]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3162]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3169]: refused connect from toychest.telisphere.com
Aug 25 16:58:33 our.web.server in.telnetd[3170]: refused connect from toychest.telisphere.com
```

Source of trace

This came from the tcpdlog on our web server some time ago and is a bit dated.

Detect was generated by

This was an intercept by tcpwrapper.

Fields:

```
Aug 25 16:58:33 our.web.server in.telnetd[3170]: refused connect from
toychest.telisphere.com
```
| Date | Local Time| Destination | Service [port] | Action |
| Source IP or resolved name |

Probability the source address was spoofed
I don't believe that the source address was spoofed. The attacker would want to have access to the shell once the connection was established.

Description of attack
This is a brute force attack against weak passwords and sloppy system administration.

Attack mechanism
This appears to be an automated script due to the relative proximity of port numbers and time stamps. Since all connection attempts are to the telnet service it would be reasonable to assume that this was a brute force password guessing attempt looking for common passwords used by careless web server administrators. Although by today's standards this is a rather loud and clumsy attack, at the time it would probably not have been noticed by a vast majority of system users or network administrators due to the short duration.

Correlations
Several other attempts have been made against this system but are usually just one or two quick connect attempts on the telnet and ftp services then the perpetrators move on to other systems.

Evidence of active targeting
I would assume that this host was actively targeted. At the time of the attack, it was the only system outside the firewall but it is also a well known system due to its function as a web server. The nature of this system is to be accessed by the general public.

Severity
(4 + 4) - (5 + 2) = 1

Criticality = 4 (web server, potentially embarrassing but no permanent loss of data or customers)
Lethality = 4 (compromise of server would result in denial of service but there is no trust relationship to the inside to be exploited)
System countermeasures = 5 (system has been hardened, patches applied)
Network countermeasures = 2 (there are measures to prevent spread of damage but nothing to help the server itself)

Defensive recommendation
This system held up well. All unnecessary services have been removed and those that are necessary are tightened down and closely watched. It would have been a more complete log, however, had the user names that were attempted been recorded. An intrusion detection system would help to monitor the traffic to the server. Additionally a set of screening acls on the router to prevent access to the telnet service from the outside world would add redundancy.

Question

This trace represents:
a) normal traffic

b) a user with “fat finger syndrome”
c) an attempt to guess account names and passwords
d) none of these

answer:c

Detect 2

```
Aug 10 05:18:27 our.web.server in.ftpd[600]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:33 our.web.server in.ftpd[601]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:36 our.web.server in.rlogind[604]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:36 our.web.server in.telnetd[602]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:36 our.web.server in.ftpd[603]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:40 our.web.server in.ftpd[606]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:46 our.web.server in.ftpd[607]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:52 our.web.server in.ftpd[608]: refused connect from elewis.ios.doi.gov
Aug 10 05:18:59 our.web.server in.ftpd[609]: refused connect from elewis.ios.doi.gov
Aug 10 05:19:05 our.web.server in.ftpd[610]: refused connect from elewis.ios.doi.gov
Aug 10 05:19:11 our.web.server in.telnetd[611]: refused connect from elewis.ios.doi.gov
Aug 10 05:35:28 our.web.server in.telnetd[644]: refused connect from elewis.ios.doi.gov
Aug 10 05:35:46 our.web.server in.rlogind[645]: refused connect from elewis.ios.doi.gov
Aug 10 05:38:55 our.web.server in.ftpd[648]: refused connect from elewis.ios.doi.gov
Aug 10 05:40:07 our.web.server in.ftpd[651]: refused connect from elewis.ios.doi.gov
Aug 10 05:41:36 our.web.server in.ftpd[652]: refused connect from elewis.ios.doi.gov
Aug 12 14:04:05 our.web.server in.ftpd[1247]: refused connect from elewis.ios.doi.gov
Aug 12 14:04:05 our.web.server in.rlogind[1248]: refused connect from elewis.ios.doi.gov
Aug 12 14:04:05 our.web.server in.ftpd[1250]: refused connect from elewis.ios.doi.gov
Aug 12 14:04:05 our.web.server in.telnetd[1246]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:43 our.web.server in.ftpd[1285]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:44 our.web.server in.ftpd[1286]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:45 our.web.server in.ftpd[1292]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:46 our.web.server in.ftpd[1287]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:46 our.web.server in.ftpd[1291]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:46 our.web.server in.ftpd[1290]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:46 our.web.server in.ftpd[1288]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1294]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1299]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1289]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1295]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1293]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:47 our.web.server in.ftpd[1296]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:49 our.web.server in.ftpd[1297]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:49 our.web.server in.ftpd[1298]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:50 our.web.server in.ftpd[1305]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:50 our.web.server in.ftpd[1300]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:50 our.web.server in.ftpd[1303]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:51 our.web.server in.ftpd[1301]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:51 our.web.server in.ftpd[1302]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:52 our.web.server in.ftpd[1304]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:52 our.web.server in.ftpd[1307]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:52 our.web.server in.ftpd[1309]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:52 our.web.server in.ftpd[1306]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:52 our.web.server in.ftpd[1308]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1310]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1311]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1312]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1315]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1314]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1316]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1317]: refused connect from elewis.ios.doi.gov
Aug 12 14:17:53 our.web.server in.ftpd[1313]: refused connect from elewis.ios.doi.gov
Aug 12 14:18:03 our.web.server in.telnetd[1318]: refused connect from elewis.ios.doi.gov
```

Source of trace

This came from the tcpdlog on our web server.

Detect was generated by
This was an intercept by tcpwrapper.

Fields:

```
Aug 25 16:58:33 our.web.server in.telnetd[602]: refused connect from
elewis.ios.doi.gov
| Date | Local Time| Destination      | Service [port]      | Action          |
|    Source IP or resolved name     |
```

Probability the source address was spoofed
Address spoofing is unlikely. The attacker would need the results of the various attempts.

Description of attack
This was a combination of service probes and brute force attacks.

Attack mechanism
The attacker probed different services that may have been provided by the server. Those that were being monitored were recorded. The attacker also ran a brute force account and password guessing attempt on the ftp service. The whole scenario was repeated two days later. Due to the time and port number sequences it would be safe to assume that this was a script being run from a possibly compromised machine trying to exploit a .gov to .gov trust relationship.

Correlations
I have no correlating evidence of my own because this machine was the only one outside the firewall at the time. Possibly supporting firewall logs are currently unavailable. This is evidence of fairly common script activity run by newbies at the time. More sophisticated tools today hide the attempts by lowering frequency or crafting packets to generate additional traffic flow to blind log based systems to the true identity of the attacking system.

Evidence of active targeting
This is a government web server. Experiences of other departments would lead to the reasonably safe conclusion that this was targeted for compromise.

Severity
(4 + 4) - (5 + 2) = 1

Criticality = 4 (web server, potentially embarrassing but no permanent loss of data or customers)
Lethality = 4 (compromise of server would result in denial of service but there is no trust relationship to the inside to be exploited)
System countermeasures = 5 (system has been hardened, patches applied)
Network countermeasures = 2 (there are measures to prevent spread of damage but nothing to help the server itself)

Defensive recommendation
This system was not compromised. All unnecessary services have been removed and security patches installed. Again, it would have been a more complete log had the user

names that were used to attempt ftp access been recorded. An ids would help to monitor the traffic to the server. Additionally a set of screening acls on the router or putting the server on a protected subnet would add redundancy.

Question
The server in the trace was:
a) scanned then compromised
b) scanned but not compromised
c) not available to the network
d) running services on generally unknown ports

answer:b

Detect 3

```
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.120" "tcp" "19" "2938" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.121" "tcp" "19" "2939" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.122" "tcp" "19" "2940" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.119" "tcp" "19" "2937" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.123" "tcp" "19" "2941" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.124" "tcp" "19" "2942" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.125" "tcp" "19" "2943" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.128" "tcp" "19" "2946" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.130" "tcp" "19" "2948" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.132" "tcp" "19" "2950" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.153" "tcp" "19" "3027" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.154" "tcp" "19" "3028" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.157" "tcp" "19" "3031" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.127" "tcp" "19" "2945" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.126" "tcp" "19" "2944" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.129" "tcp" "19" "2947" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.131" "tcp" "19" "2949" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.152" "tcp" "19" "3026" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.155" "tcp" "19" "3029" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.156" "tcp" "19" "3030" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.158" "tcp" "19" "3032" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.166" "tcp" "19" "3040" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.167" "tcp" "19" "3041" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.171" "tcp" "19" "3045" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.170" "tcp" "19" "3044" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.174" "tcp" "19" "3048" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.175" "tcp" "19" "3049" " len 60"
" 0:03:46" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.177" "tcp" "19" "3051" " len 60"
.
.
.
" 0:03:49" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.174" "tcp" "19" "4819" " len 60"
" 0:03:49" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.178" "tcp" "19" "4823" " len 60"
```

Source of trace
Our network.

Detect was generated by
This data was extracted from our Firewall log.

Fields:
```
" 0:03:49" "qfe0" "drop" "sunrpc" "193.89.57.8" "my.net.178" "tcp" "19" "4823" " len 60"
```
| time | intfc | action | service | src addr | dest addr | proto | rule| src port |info

Probability the source address was spoofed

It is unlikely that the address was spoofed because the attacker would like to get the info on the RPC services that this scan would find.

Description of attack
This is a quick but noisy scan to locate machines providing remote procedure call services. Unfortunately for this fellow the attack occurred in the middle of the night (local time) when there wasn't much for traffic. This scan stuck out like the Washington monument.

Attack mechanism
Due to the large volume of machines probed (194 in the full log) and the short duration (3 to 4 seconds) and the sequential nature of the source port numbers it would be safe to assume that this was a script. The purpose of connecting to the portmapper is to get the information it is designed to hand out to base a more stealthy and custom attack or to exploit the portmapper itself for root access.

Correlations
This appears to have been a hit and run from this site however later in the month two more scans were recorded. They also had the high volume, random machine, and sequential source port indicators of a script being run. There are thirteen documented vulnerabilities revolving around RPC services and portmapper in the CVE database from early 1999 on and another eight candidates being reviewed.

Evidence of active targeting
Specific machines were not actively targeted. The full trace includes attempts to nonexistant systems and systems that do not run portmapper.

Severity
(2 + 5) - (5 + 5) = -3

Criticality = 2 (I'll give this a two because some of the targeted machines do run portmapper)
Lethality = 5 (the intent was to get root access)
System countermeasures = 5 (systems have had patches applied, others no services)
Network countermeasures = 5 (there is a firewall that does not permit access to these services)

Defensive recommendation
Current defenses, applied security patches from vendors and firewall prohibiting inbound rpc traffic, are sufficient.

Question
Based on the above trace:
a) the user of machine 193.89.57.8 deserves a productivity raise
b) my.net is being scanned for trojans
c) my.net is being probed for open portmapper services
d) all of the above

answer:c

Detect 4

```
[Sun May  7 12:20:10 2000] access to /usr/local/etc/httpd/cgi-bin/phf failed for
fs01.yz.yamagata-u.ac.jp, reason: Client denied by server configuration
```

Source of trace
Our web server.

Detect was generated by
This detect was extracted from our web server error log.

Probability the source address was spoofed
It is unlikely that the address is spoofed.

Description of attack
This attack was an attempt to execute the example program phf that is generally installed with web servers.

Attack mechanism
The attacker passes parameters through the web connection to execute phf. It can often be exploited to run commands on the attacker's behalf, sometimes with above normal privileges. We haven't had one of these for a long time but this type of attack is still making the rounds and is even on a top ten list as noted by http://www.sans.org/y2k/051400.htm.

Correlations
We had an event like this quite some time ago in an attempt to cat the passwd file. This seems to be a lone event as out other server wasn't accessed. It is showing up regularly though, other examples can be found at: http://www.sans.org/y2k/051400.htm. It is described in CVE-1999-0067 and CERT:CA-96.06.

Evidence of active targeting
This is a primary web server. It has a partner server that works silently in the background. Since the primary server was attacked but the other was not I would say that it was actively targeted.

Severity
(4 + 4) - (5 + 2) = -2

Criticality = 4 (This is our web server and there are no trust or infrastructure relationships that can be abused)
Lethality = 4 (the intent was to get root access and that could have cost us the use of the web server)
System countermeasures = 5 (systems have had patches applied, httpd configured restrictively)
Network countermeasures = 2 (there is a firewall that does not permit access from the server limiting the potential damage)

Defensive recommendation
With the httpd being properly configured against these types of attacks the host based intrusion prevention seems to be sufficient at this time.

Question
This is:
a) an attempt to talk to a ddos slave
b) a probe to determine OS make and model

c) an example of bad ftp grammar
d) failed attempt to take advantage of a poor installation

answer:d

Detect 5

```
2000/04/13 13:12:04.0296,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:13:18.0859,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:14:22.0250,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:15:25.0328,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:16:27.0578,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:17:27.0687,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:18:42.0328,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:19:47.0250,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:20:48.0156,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:21:49.0375,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:22:54.0921,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:24:05.0734,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:25:13.0468,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
2000/04/13 13:26:39.0125,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
```

Source of trace
Our network.

Detect was generated by
Network-1 CyberWall Plus on our ancillary web server. Unfortunately the details of the detect are not exportable nor available for cut and paste. They can be viewed online. Each line in the log represents a violation of a rule that prohibits more than 20 simultaneous connects.

Fields:

```
2000/04/13 13:26:39.0125,Untrusted NIC,Transport,TCP,TCP Port Scan,Any
Node,198.110.199.41,<DNS Lookup Failed>,Local Machine,my.net1.5,our-server
```

| date - time | interface | detected at | proto | type of attack | log from where | source ip | resolved name | log to | dest ip addr | resolved name

the log from where and to where relate to the rule base. In this case, traffic *from any machine* directed *to me* is logged if it triggers a rule such as too many simultaneous connections.

Probability the source address was spoofed
Not likely since the attacker would be interested in knowing the results. It is more likely that the source machine had a compromised account.

Description of attack
This is an attempt to determine either the operating system based on responses to queries of services or an attempt to probe for active ports.

Attack mechanism
Based on the fact that each line in the log represents a violation of a rule that prohibits more than 20 simultaneous connections and the time spacing, I would say that this was an automated attack. The lack of details in the log prevents me from determining which particular script it may have been. Generally, the attacker would send syn packets to a bunch of ports. Those that had daemons listening on them would "ack" back. The well known port number that responded would dictate the next step to compromising the system with assorted "toolkits" designed for different services.

Correlations
There were no other attempts from this address to this machine. There were also no other attempts within a reasonable enough time frame to suggest an obvious multiple source machine coordinated probe of this machine. Port scanning is a very common method to determine vulnerabilities on known systems.

Evidence of active targeting
There were no correlating entries in the firewall log or on the other web server so this was either a very lucky strike in a wide spread pattern or the knowledge of this machine pre-existed the scan attempt.

Severity
(4 + 3) - (5 + 2) = 0

Criticality = 4 (This is our ancillary web server and isn't critical to our infrastructure)
Lethality = 3 (the intent was to gain access and that could have cost us the use of the web server)
System countermeasures = 5 (systems have had patches applied, httpd configured restrictively)
Network countermeasures = 2 (there is a firewall that does not permit access from the server limiting the potential damage)

Defensive recommendation
The host based intrusion detection/prevention software on this system is adequate for this type of attack.

Question
This is:
a) a secure shell scan
b) a probe for open ports
c) an ftp transfer from an unauthorized host
d) a bad day in domain name services

answer:b

Detect 6

```
" 2:20:05"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net1.2"  "tcp"  "22549"  " len 44"
" 2:20:14"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.1"  "tcp"  "26487"  " len 44"
```

```
" 2:20:14"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.2"  "tcp"  "26488"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.22"  "tcp" "26551"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.25"  "tcp" "26489"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.21"  "tcp" "26609"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.20"  "tcp" "26640"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.19"  "tcp" "26643"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.17"  "tcp" "26653"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.16"  "tcp" "26694"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.15"  "tcp" "26742"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.13"  "tcp" "26745"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.7"  "tcp"  "26785"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.6"  "tcp"  "26837"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.5"  "tcp"  "26841"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.3"  "tcp"  "26844"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.50" "tcp"  "26883"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.47" "tcp"  "26889"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.46" "tcp"  "26928"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.44" "tcp"  "26934"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.24" "tcp"  "26490"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.42" "tcp"  "26979"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.41" "tcp"  "26996"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.23" "tcp"  "26491"  " len 44"
" 2:20:15"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net2.39" "tcp"  "27002"  " len 44"

.

.

.
" 2:21:09"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net5.254" "tcp" "10312"  " len 44"
" 2:21:09"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net5.253" "tcp" "10313"  " len 44"
" 2:21:09"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net5.252" "tcp" "10314"  " len 44"
```

Source of trace
Our network.

Detect was generated by
Check Point Firewall

Fields:
```
" 2:21:09"  "qfe0"  "drop"  "imap2"  "12.3.72.138"  "my.net5.252" "tcp" "10314"  " len 44"
```
| time | intfc | action | service | source address| dest address|proto|source port| info

Probability the source address was spoofed
It is more likely that this is a throw away host. The large volume of attempts in such a short time will undoubtly gather attention so it is a get in, get info, and get away operation.

Description of attack
This is an attack against the imap service. There are many exploits available against imap as documented in CVE, CERT, BUGTRAQ, and many vendor lists.

Attack mechanism
This is a script probing our entire local network. It is probable that they are mapping networks by sifting through host unreachable verses port unreachable messages. It is also a method for looking for a specific type of system to try out their latest gimmick. They can determine this by the response they get from the imap port on some systems.

Correlations
We seem to get hit with these every other month or so. Curiously, it always seems to happen at the end of the month, the 23rd through the 28th. This was the most extensive to date, over 1000 addresses probed.

Evidence of active targeting
They weren't targeting anything below the network level. It is the equivalent of opening a fire hose on a house to check for open windows. The whole dang thing gets wet.

Severity
(2 + 2) - (4 + 5) = -5

Criticality = 2 (Most targets were desktop units and unlikely to be running imap)
Lethality = 2 (the intent seems to be to probe for existence of machines)
System countermeasures = 4 (Assuming that users or other admins haven't installed imap services)
Network countermeasures = 5 (there is a firewall that does not permit access to the imap service)

Defensive recommendation
Defenses provided by the firewall are sufficient, putting the probes into the bit bucket not even allowing responses.

Question
This is an example of:
a) an imap scan
b) a syn flood
c) a buffer overflow attack
d) an exploit of bad karma

answer:a

Detect 7

```
"10:37:26"  "qfe0"  "drop"  "http"  "213.219.19.148"  "my.net2.1" "tcp" "62932"  " len 48"
"10:59:17"  "qfe0"  "drop"  "http"  "213.219.19.148"  "my.net3.1" "tcp" "63606"  " len 48"
"11:42:46"  "qfe0"  "drop"  "http"  "213.219.19.148"  "my.net5.1" "tcp" "63505"  " len 48"
```

Source of trace
This was recorded on our network.

Detect was generated by
Check Point Firewall

```
"11:42:46"  "qfe0"  "drop"  "http"  "213.219.19.148"  "my.net5.1" "tcp" "63505"  " len 48"
| time      | intfc  | action | service | source address| dest address|proto|source port| info
```

Probability the source address was spoofed
The probability that the address was spoofed is unlikely because the attacker would be interested gathering the addresses of the servers.

Description of attack
This was just a brief visit by someone looking for new web servers to exploit.

Attack mechanism
There are many, many attacks against web servers. CVE lists over 40 and is evaluating several dozen more. Exploiting these weaknesses can give the attackers a repository for warez, a launching point for further data gathering or ddos zombies or could be politically motivated due to our domain.

Correlations
The large number exploits available for the various types of servers and the fact that web servers are most useful when people can get to them means that they will continually attract unwanted visitors. These kinds of low key probes will probably always be a part of a web administrators life.

Evidence of active targeting
This was a probe for potential web servers so there wasn't active targeting beyond the network level. The prober just used the same host address and rotated through a class address. There were probably many more probes sent out by this machine that I didn't see due to routing.

Severity
(2 + 2) - (5 + 5) = -6

Criticality = 2 (This is a probe for potential web servers, it's not directed to known machines)
Lethality = 2 (the intent was to discover not gain access to a web server)
System countermeasures = 5 (systems have had patches applied, httpd isn't running)
Network countermeasures = 5 (there is a firewall that does not permit access from the internet on this service)

Defensive recommendation
The current defense of blocking http requests directed to random machines is sufficient.

Question
This is:
a) a probe for web servers
b) a probe for domain name servers
c) a probe for trojans
d) a network mapping attempt

answer:a

Detect 8

```
" 0:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net2.0"  "udp"  "1036"
" 0:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.0"  "udp"  "1036"
" 0:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net4.0"  "udp"  "1036"
" 0:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.0"  "udp"  "1036"
" 9:08:41"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net2.35"  "udp"  "1024"
" 9:08:41"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.35"  "udp"  "1024"
" 9:08:41"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net4.35"  "udp"  "1024"
" 9:31:37"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.37"  "udp"  "1024"
" 9:31:37"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.37"  "udp"  "1024"
"10:28:56"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.42"  "udp"  "1024"
"12:23:44"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net2.52"  "udp"  "1024"
"12:23:44"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.52"  "udp"  "1024"
"12:23:44"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net4.52"  "udp"  "1024"
```

```
"12:23:44"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.52"  "udp"  "1024"
"20:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net2.92"  "udp"  "1024"
"20:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.92"  "udp"  "1024"
"20:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net4.92"  "udp"  "1024"
"20:02:43"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.92"  "udp"  "1024"
"22:31:53"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.105"  "udp"  "1024"
"22:31:53"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.105"  "udp"  "1024"
"22:43:23"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net3.106"  "udp"  "1024"
"22:43:23"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net2.106"  "udp"  "1024"
"22:43:23"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net4.106"  "udp"  "1024"
"22:43:23"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.106"  "udp"  "1024"
```

Source of trace
This was captured on our network.

Detect was generated by
Check Point Firewall

Fields:
```
"22:43:23"  "qfe0"  "drop"  "snmp"  "207.96.37.201"  "my.net5.106"  "udp"  "1024"
```
| time | intfc | action | service| source address| dest address | proto | source port

Probability the source address was spoofed
It is unlikely that the address was spoofed.

Description of attack
This is an information gathering attack.

Attack mechanism
The attacker probes for snmp daemons that were perhaps left with default access strings after install. Many devices come with snmp agents built in to them. There were a couple of attempts at the network address hoping to get a mass response from all devices that use that as the broadcast address. The resulting replies will potentially give the attacker a lot of additional information.

Correlations
Perhaps not as widely used as some other services it is often overlooked when it comes time to cleanup and install. There are vulnerabilities listed on both the CVE and BUGTRAQ boards.

Evidence of active targeting
I believe that these addresses were randomly selected. The repeats were perhaps attempts to ensure time outs and packet losses did not occur. Repeat scans to systems have shown up in other traces.

Severity
(2 + 4) - (5 + 5) = -4

Criticality = 2 (This is a probe for potential web servers, it's not directed to known machines)
Lethality = 4 (access gained through this service would very costly)
System countermeasures = 5 (systems have patches applied, snmp agents aren't running)
Network countermeasures = 5 (there is a firewall that does not permit access from the internet on this service)

Defensive recommendation
These packets are blocked at the firewall. If there is a need to allow snmp into your network it should be closely monitored.

Question
In this trace the attacker was :
a) successful
b) transferring a domain zone
c) looking for agents
d) none of the above

answer: c

Detect 9

```
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.10"  "udp"  "1746"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.13"  "udp"  "1749"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.20"  "udp"  "1752"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.21"  "udp"  "1751"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.22"  "udp"  "1755"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.23"  "udp"  "1758"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.24"  "udp"  "www-ldap-gw"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.25"  "udp"  "1762"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.26"  "udp"  "1765"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net2.27"  "udp"  "1767"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.12"  "udp"  "1770"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.15"  "udp"  "1772"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.18"  "udp"  "1777"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.24"  "udp"  "1776"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.58"  "udp"  "1780"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.178"  "udp"  "1786"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.87"  "udp"  "1790"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net4.66"  "udp"  "1784"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.21"  "udp"  "1795"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.22"  "udp"  "1797"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.237"  "udp"  "1799"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.3"  "udp"  "1802"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.27"  "udp"  "1801"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.34"  "udp"  "1805"
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.42"  "udp"  "1807"
```

Source of trace
Our network.

Detect was generated by
Check Point Firewall

Fields:
```
" 1:36:18"  "qfe0"  "drop"  "domain-udp"  "144.92.98.76"  "my.net5.42"  "udp"  "1807"
| time      | intfc | action | service       | source address| dest address| proto| source port
```

Probability the source address was spoofed
Unlikely since they were looking for domain servers. More likely they are using a "throw away" host.

Description of attack
This is a quick sweep for domain name servers.

Attack mechanism
By locating domain name servers the attackers can come back at a later time and ply a couple of exploits against them. Cracking open a dns saves them a lot of detection avoidance down the road while trying to map networks because a lot of the information they are looking for is probably in the dns. Controlling a dns also gives the attackers a way to masquerade other attacks by poisoning neighbor dns systems.

Correlations
Domain name servers will be juicy targets because they are part of the infrastructure. Being in control of the map makers lets you direct traffic where ever you need it. Their attractiveness as targets subjects them to denial of service attacks as well as exploiting vulnerabilities as described in CERT, CVE, BUGTRAQ and many vendor information sources.

Evidence of active targeting
There was no active host targeting. It was a sweep of mostly low addresses where a lot of system administrators put their servers.

Severity
(2 + 4) - (5 + 5) = -4

Criticality = 2 (This is a probe for potential web servers, it's not directed to known machines)
Lethality = 4 (access gained through this service would very costly)
System countermeasures = 5 (systems have patches applied, named isn't running)
Network countermeasures = 5 (there is a firewall that does not permit access from the internet on this service)

Defensive recommendation
This probe was blocked by the firewall. DNS machines should get careful scrutiny and diligent maintenance to protect network neighbors.

Question
This attacker was looking for:
a) dns servers
b) ftp servers
c) coffee servers
d) ice cream

answer: a

Detect 10

```
"28Apr2000"  "21:11:55"  "qfe0"  "drop" "216.88.159.118"  "my.net5.0"  "icmp"  " icmp-type 11
icmp-code 0"
"29Apr2000"  " 3:45:44"  "qfe0"  "drop" "216.88.159.118"  "my.net4.0"  "icmp"  " icmp-type 11
icmp-code 0"
"29Apr2000"  "12:07:42"  "qfe0"  "drop" "216.88.159.118"  "my.net3.0"  "icmp"  " icmp-type 11
icmp-code 0"
"29Apr2000"  "22:36:14"  "qfe0"  "drop" "216.88.159.118"  "my.net3.0"  "icmp"  " icmp-type 11
icmp-code 0"
"30Apr2000"  " 1:44:07"  "qfe0"  "drop" "216.88.159.118"  "my.net2.0"  "icmp"  " icmp-type 11
icmp-code 0"
"30Apr2000"  " 6:01:43"  "qfe0"  "drop" "203.12.48.138"  "my.net5.0"  "icmp"   " icmp-type 3
icmp-code 9"
```

```
"30Apr2000"  " 6:25:31"  "qfe0"  "drop" "216.88.159.118"  "my.net2.0"  "icmp"  " icmp-type 11
icmp-code 0"
"30Apr2000"  " 8:54:14"  "qfe0"  "drop" "203.12.48.138"  "my.net4.0"  "icmp"   " icmp-type 3
icmp-code 9"
"30Apr2000"  "10:52:40"  "qfe0"  "drop" "216.88.159.118"  "my.net4.0"  "icmp"  " icmp-type 11
icmp-code 0"
"30Apr2000"  "11:12:21"  "qfe0"  "drop" "203.12.48.138"  "my.net3.0"  "icmp"   " icmp-type 3
icmp-code 9"
"30Apr2000"  "20:56:43"  "qfe0"  "drop" "203.12.48.138"  "my.net3.0"  "icmp"   " icmp-type 3
icmp-code 9"
"30Apr2000"  "22:54:59"  "qfe0"  "drop" "203.12.48.138"  "my.net3.0"  "icmp"   " icmp-type 3
icmp-code 9"
"1May2000"  "14:55:39"  "qfe0"  "drop"  "216.88.159.118"  "my.net5.0"  "icmp"  " icmp-type 11
icmp-code 0"
"1May2000"  "18:19:16"  "qfe0"  "drop"  "216.88.159.118"  "my.net3.0"  "icmp"  " icmp-type 11
icmp-code 0"
"1May2000"  "14:55:39"  "qfe0"  "drop"  "216.88.159.118"  "my.net5.0"  "icmp"  " icmp-type 11
icmp-code 0"
"1May2000"  "18:19:16"  "qfe0"  "drop"  "216.88.159.118"  "my.net3.0"  "icmp"  " icmp-type 11
icmp-code 0"
"3May2000"  " 1:49:09"  "qfe0"  "drop"  "216.88.159.118"  "my.net4.0"  "icmp"  " icmp-type 11
icmp-code 0"
"3May2000"  " 3:05:42"  "qfe0"  "drop"  "216.88.159.118"  "my.net2.0"  "icmp"  " icmp-type 11
icmp-code 0"
"3May2000"  "10:06:29"  "qfe0"  "drop"  "216.88.159.118"  "my.net4.0"  "icmp"  " icmp-type 11
icmp-code 0"
```

Source of trace
This was collected on our network.

Detect was generated by
Check Point Firewall.

Fields:
```
"3May2000"  "10:06:29"  "qfe0"  "drop"  "216.88.159.118"  "my.net4.0"  "icmp"  " icmp-type 11
icmp-code 0"
```
| date | time | intfc | action | source address | dest address | proto | info

Probability the source address was spoofed
It is unlikely that the source address was spoofed. The attacker used other means to avoid detection.

Description of attack
This is an example of a low and slow probe combined with pokes at the .0 address to see if any old BSD style machines would answer.

Attack mechanism
By spacing the probes out over a couple of days and paced a couple to several hours apart the attacker on 216 will avoid detects on systems that monitor frequency. The probes to the .0 address were meant to elicit a broadcast response from older systems. 203. seems to be repeating the same probes. It may be another attempt from a different angle by the same person or a partner.

Correlations
This approach was designed to avoid detects like that exhibited in the Detect 5 section of this document. With more and more IDS's checking the frequency of visits this is becoming the prefered approach. The black hats have all the time they need as long as they aren't detected. Patience and automation are on their side.

Evidence of active targeting
The targeting is at the network level not at specific hosts.

Severity
(2 + 3) - (5 + 5) = -5

Criticality = 2 (This is a mapping attempt, it's not directed to known machines)
Lethality = 3 (the intent was to discover active machines giving the attacker new targets)
System countermeasures = 5 (systems have had patches applied, most don't answer icmp)
Network countermeasures = 5 (there is a firewall that does not permit access from the internet on this service)

Defensive recommendation
The current firewall configuration to block incoming icmp packets will prevent these kinds of mapping attempts.

Question
This is:
a) a file transfer log
b) a port scan
c) a mapping attempt
d) a dns zone transfer

answer:c