Global Information Assurance Certification Paper

SANS GIAC Certified Intrusion Analyst Practical
Fleet Information Warfare September 11-14, 2000
Arnold C. Llamas, Systems Engineer

Detect #1

1. Source of trace:

- Fleet Information Warfare Center CND (Computer Network Defense)
- IP addresses were sanitized.
- This trace is from a class B network. A firewall and a screening router are the known defensive measures in place. The target of this attack was the class B network's outer router.

15:31:35.635903 guarajuba.telemar-ba.net.br.65044 > x.y.180.87.55763: R 0:0(0) ack 1 win 0
15:32:57.006145 guarajuba.telemar-ba.net.br.48269 > x.y.52.120.24517: R 0:0(0) ack 1 win 0
15:36:55.285727 guarajuba.telemar-ba.net.br.41834 > x.y.229.116.36451: R 0:0(0) ack 1 win 0
15:39:27.919194 guarajuba.telemar-ba.net.br.51089 > x.y.132.89.34229: R 0:0(0) ack 1 win 0
15:41:30.808180 guarajuba.telemar-ba.net.br.60944 > x.y.128.100.36012: R 0:0(0) ack 1 win 0
15:41:43.816802 guarajuba.telemar-ba.net.br.53153 > x.y.106.8.8006: R 0:0(0) ack 1 win 0
15:41:55.042999 guarajuba.telemar-ba.net.br.50357 > x.y.187.62.30105: R 0:0(0) ack 1 win 0
15:42:50.708963 guarajuba.telemar-ba.net.br.21392 > x.y.27.23.55823: R 0:0(0) ack 1 win 0
15:43:06.639423 guarajuba.telemar-ba.net.br.33582 > x.y.59.95.64395: R 0:0(0) ack 1 win 0
15:43:20.248203 guarajuba.telemar-ba.net.br.3014 > x.y.64.21.21911: R 0:0(0) ack 1 win 0
15:43:58.691761 guarajuba.telemar-ba.net.br.43583 > x.y.247.68.9932: R 0:0(0) ack 1 win 0
15:44:12.423295 guarajuba.telemar-ba.net.br.35792 > x.y.225.104.47462: R 0:0(0) ack 1 win 0
15:44:49.265837 guarajuba.telemar-ba.net.br.17414 > x.y.6.103.13550: R 0:0(0) ack 1 win 0
15:45:26.134307 guarajuba.telemar-ba.net.br.46789 > x.y.173.9.15264: R 0:0(0) ack 1 win 0
15:45:59.567526 guarajuba.telemar-ba.net.br.48392 > x.y.8.44.17930: R 0:0(0) ack 1 win 0
15:46:24.548117 guarajuba.telemar-ba.net.br.37805 > x.y.67.6.12023: R 0:0(0) ack 1 win 0
15:47:58.864599 guarajuba.telemar-ba.net.br.58831 > x.y.47.109.56467: R 0:0(0) ack 1 win 0
15:49:22.411076 guarajuba.telemar-ba.net.br.34265 > x.y.153.49.62752: R 0:0(0) ack 1 win 0
15:53:47.466652 guarajuba.telemar-ba.net.br.16195 > x.y.171.87.32577: R 0:0(0) ack 1 win 0
15:54:23.541187 guarajuba.telemar-ba.net.br.2812 > x.y.55.104.48770: R 0:0(0) ack 1 win 0
20:51:43.482879 guarajuba.telemar-ba.net.br.51544 > x.y.101.14.40746: R 0:0(0) ack 1 win 0
20:51:47.018721 guarajuba.telemar-ba.net.br.28767 > x.y.128.32.26267: R 0:0(0) ack 1 win 0
20:51:51.895699 guarajuba.telemar-ba.net.br.5990 > x.y.155.50.11788: R 0:0(0) ack 1 win 0
20:52:04.857340 guarajuba.telemar-ba.net.br.18180 > x.y.187.122.20361: R 0:0(0) ack 1 win 0
20:52:20.162941 guarajuba.telemar-ba.net.br.15384 > x.y.12.49.42460: R 0:0(0) ack 1 win 0
20:53:55.078436 guarajuba.telemar-ba.net.br.64179 > x.y.222.114.53179: R 0:0(0) ack 1 win 0
20:54:16.789084 guarajuba.telemar-ba.net.br.53592 > x.y.25.77.47273: R 0:0(0) ack 1 win 0
20:54:29.176286 guarajuba.telemar-ba.net.br.45801 > x.y.3.113.19267: R 0:0(0) ack 1 win 0
20:54:36.241253 guarajuba.telemar-ba.net.br.247 > x.y.57.21.55845: R 0:0(0) ack 1 win 0

2. Detect was generated by:

- Tcpdump v3.5
- Log format
  - 20:54:36.241253 guarajuba.telemar-ba.net.br.247 > x.y.57.21.55845: R 0:0(0) ack 1 win 0
  - A B C D E F G
  - A – timestamp (hh:mm:ss.fraction_of_a_second)
  - B – source address.port

- C – destination address.port
- D – Flags (TCP flags in this case)
- E – Beginning sequence number:ending sequence number (bytes)
- F – Acknowledgement number
- G – Window size

- The relevant information for this line follows:
  - Timestamp – 20:54:36.241253
  - Source address.port – 200.223.0.126.247
  - Destination address.port – x.y.57.21.55845
  - Flags – **R**eset (TCP flags can be **U**rgent, **A**cknowledge, **P**ush, **R**eset, **S**yn, or **F**in)
  - Beginning sequence number:Ending sequence number (bytes) – 0:0(0)
  - **Ack**nowledgement number – the next sequence number that the sender of the ACK expects to receive. It's set to 1 in this trace.
  - Window size – the number of bytes in the receiver's buffer. It's set to 0 in this case.

3. Probability the source address was spoofed (probably spoofed, probably not spoofed, or 3rd party):

- Very low. This probe occurred over a period of over 5 hours to many machines behind this class B network's defenses for recon purposes. Information gathered during this pre-attack phase would need to be sent to a valid address for collection.

4. Description of attack:

- This is a low and slow reset scan occurring over 5 hours against the target network's outer router. Only 29 packets were sent during this timeframe. The most likely cause of this activity is to inversely map the network for hosts that are *not* alive.

5. Attack mechanism:

- The attacker sends a RST packet to the target network with the following TCP header information:
  - Reset flag set
  - Beginning and ending sequence number set to 0 with 0 bytes.
  - Acknowledgement number set to 1.
  - Window size set to 0.
- The router examines this packet and simply responds with an "icmp: <dest ip address> unreachable" if the address doesn't exist. This information tells the attacker which hosts *don't* exist on that network.
- This method is popular because it's difficult to detect (low and slow), and it can be even more difficult to detect if conducted from several disparate IP addresses combined with the low and slow approach.
- It's interesting to note that an ACK number of 674719802 used to be the norm for this type of scan. However, this ACK number became its telltale signature and recent reset scans have used different ACK numbers to evade detection (see trace above and "Correlations" below).

6. Correlations:

- The following traces from Dave Goldsmith and Steve Carey also mention reset scans that they detected on their networks. However, the ACK numbers on these traces do *not* use the telltale ACK number of 674719802 (to avoid easy detection).
- http://www.sans.org/y2k/081000.htm - Dave Goldsmith
- http://www.sans.org/y2k/070400.htm - Steve Carey
- http://www.sans.org/y2k/071200.htm - Steve Carey
- Reset scans were also detected over 200 times this year here and at other military sites.

- An nslookup on guarajuba.telemar-ba.net.br returned this source address: 200.223.0.126. Consequently, a whois lookup on 200.223.0.126 revealed that the traffic originated from the Brazilian Research Network (RNP). Activity from the Brazilian Research Network is quite common in our database and at SANS GIAC. The following is just a sample of traffic originating from RNP: reset scans, NetBIOS scans, rpc dump, and DNS zone transfers.
  - http://www.sans.org/y2k/020100.htm – rpc dump
  - http://www.sans.org/y2k/022000.htm – rpc dump
  - http://www.sans.org/y2k/022900.htm – rpc dump
  - http://www.sans.org/y2k/052700-2100.htm – probe for DNS
  - http://www.sans.org/y2k/052800-1130.htm – NetBIOS probe
  - http://www.sans.org/y2k/053100-1100.htm – DNS version
  - http://www.sans.org/y2k/060600-1200.htm – IMAP probe
  - http://www.sans.org/y2k/071000.htm – rpc dump
  - http://www.sans.org/y2k/071600.htm – reset scan
  - http://www.sans.org/y2k/081800.htm – ACK scan
  - http://www.sans.org/y2k/082000.htm – ACK scan

7. Evidence of active targeting:

- Yes. This traffic was directed towards a network for the sole purpose of inverse mapping, and these crafted packets came from the same host over a period of 5 hours. Although these packets are not targeted towards every host, it's clear that the attacker wishes to inversely map the x.y.0.0 class B network and evade detection using the low and slow approach.

8. Severity:

- (System criticality + Lethality)
  – (System Countermeasures + Network Countermeasures) = Severity
- Target network included DMZ routers/firewall.
- System criticality = 5 (outer screening router)
- Lethality = 2 (recon scan)
- System countermeasures = 5 (System fully patched)
- Network countermeasures = 5 (restrictive firewall)
- Severity = (5 + 2) – (5 + 5) = ***-3***

9. Defensive recommendation:

- Closely monitor traffic for evidence of low and slow packets with the above telltale signature. Also monitor traffic for proof of lone packets with the Reset flag set without any other associated activity (such as a SYN).
- Modify intrusion detection filters to look for packets arriving at the rate of 5(or less)/hour with the telltale signature (Reset flag set, nonsensical ACK flags).
- Patience is needed here since the attackers may be willing to send only a handful of packets per day to a network and wait a long time to map out an entire network.
- Set an ACL on the screening router to deny outgoing ICMP traffic from the router.
  - deny icmp x.y.z.1 0.0.0.0 any host-unreachable

10. Multiple choice question based on trace:
0

- What flags are set for a reset scan?
  A. Reset
  B. Push
  C. Acknowledge
  D. B & C
  E. A & C
  Answer: E

Detect #2

1. Source of trace:

- Fleet Information Warfare Center CND (Computer Network Defense) database entry dated September 2, 2000.
- IP addresses have been sanitized.

Sep 2 10:15:40 icstf-gw40 kernel: securityalert: packet denied by local screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.1 dstport=137
Sep 2 10:15:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.2 dstport=137
Sep 2 10:16:04 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.3 dstport=137
Sep 2 10:16:16 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.4 dstport=137
Sep 2 10:16:28 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.5 dstport=137
Sep 2 10:16:40 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.6 dstport=137
Sep 2 10:16:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.7 dstport=137
Sep 2 10:16:53 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.7 dstport=137
Sep 2 10:16:55 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.7 dstport=137
Sep 2 10:17:04 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.8 dstport=137
Sep 2 10:17:16 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.9 dstport=137
Sep 2 10:17:28 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.10 dstport=137
Sep 2 10:17:31 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.10 dstport=137
Sep 2 10:17:40 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.11 dstport=137
Sep 2 10:17:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.12 dstport=137
Sep 2 10:18:04 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.13 dstport=137
Sep 2 10:18:16 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.14 dstport=137
Sep 2 10:18:28 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.15 dstport=137

Sep 2 10:18:40 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.16 dstport=137
Sep 2 10:18:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.17 dstport=137
Sep 2 10:19:04 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.18 dstport=137
Sep 2 10:19:16 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.19 dstport=137
Sep 2 10:19:28 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.20 dstport=137
Sep 2 10:19:40 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.21 dstport=137
Sep 2 10:19:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.22 dstport=137
Sep 2 10:20:04 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.23 dstport=137
Sep 2 10:20:16 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.24 dstport=137
Sep 2 10:20:28 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.25 dstport=137
Sep 2 10:20:40 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.26 dstport=137
Sep 2 10:20:52 icstf-gw40 kernel: securityalert: no match found in forward screen: UDP if=eb0 srcaddr=206.10.93.22 srcport=137 dstaddr=x.y.z.27 dstport=137

2. Detect was generated by:

- Gauntlet v4.2 firewall log
- Log format
  - <timestamp> <hostname> <kernel_or_process_warnings_and_errors>
- The relevant information for this trace follows:
  - Source address – 206.10.93.22
  - Destination address – icstf-gw40
  - Protocol – UDP
  - Source port – 137 (NetBIOS)
  - Destination port – 137 (NetBIOS)
  - Interface – eb0 (packet screening rules were violated on this interface)

3. Probability the source address was spoofed (probably spoofed, probably not spoofed, or 3rd party):

- Very low. This is a recon probe of a network. The attacker would need the information returned to a valid address for information gathering purposes, so the source address is probably not spoofed.

4. Description of attack:

- This is an automated recon probe of a network for machines running NetBIOS on port 137.
- A worm named network.vbs spreads through open shares on Windows machines.
- This is a probe of the x.y.z network for NetBIOS name server (port 137). Nbtscan is an example of a NetBIOS scanner. This attack could have used nbtscan to probe the destination network for the NetBIOS name service, and it could have originated from a UNIX box since nbtscan's documentation states that root access is needed on a UNIX box to use local port 137 as the source port. I ran nbtscan on our local network and observed similar results.

5. Attack mechanism:

- This is definitely a stimulus targeting NetBIOS name service on port 137.
- This attack sends a NetBIOS "NODE STATUS REQUEST" (RFC 1001/1002) to each address and awaits a NetBIOS "NODE STATUS RESPONSE" (RFC 1001/1002). If the tool used was nbtscan, each address that responds will send its IP address, NetBIOS computer name, username, and MAC address (http://www.abb.aha.ru/software/nbtscan.html). An attacker can use this information to exploit weaknesses on machines running Windows 95/98/NT (for file/print sharing). Machines running Samba on UNIX will also be vulnerable.
- There is nothing in the CVE index at cve.mitrg.org regarding NetBIOS scanners, however if the scan above *had* returned evidence of machines running NetBIOS, an attacker could have exploited the following CVEs:
  - CVE-1999-0153 – Windows 95/NT out of band (OOB) data denial of service through NetBIOS port, aka WinNuke.
  - CVE-1999-0288 – Denial of service in WINS with malformed data to port 137 (NetBIOS Name Service).
  - CVE-1999-0810 – Denial of service in Samba NetBIOS name service daemon (nmbd).

6. Correlations

- This detect was attributed to Fleet Information Warfare Center's CND database. The actual destination address has been sanitized.
- A search of the source IP network address, 206.10.93.X, against FIWC's CND database and SANS GIAC shows no previously reported activity from this class C network. A whois lookup on 206.10.93.x reveals that the network belongs to Polaris Telecom.
- This type of scan has increased since April 2000 (from SANS GIAC). April 29, 2000 at SANS GIAC had one interesting trace that indicated traffic from France, China, India, Canada, and the US.
- Additionally, this type of traffic occurred at least 75 times on our database this year.
- NetBIOS scanning was also found at SANS GIAC
  - http://www.sans.org/y2k/122399.htm
  - http://www.sans.org/y2k/122999-1230.htm
  - http://www.sans.org/y2k/081200-1300.htm
  - http://www.sans.org/y2k/081300.htm
  - http://www.sans.org/y2k/081900.htm

7. Evidence of active targeting:

- Active targeting is clearly evident by the general scan of an entire network and port 137 (NetBIOS). This is a deliberate attempt to find Windows 95/98/NT machines and boxes running Samba.

8. Severity:

- (System criticality + Lethality)
  – (System Countermeasures + Network Countermeasures) = Severity
- Target network included DMZ routers/firewall.
- System criticality = 5 (firewall, DMZ, outer screening router)
- Lethality = 2 (recon scan)
- System countermeasures = 5 (system patched and NetBIOS not running)
- Network countermeasures = 5 (restrictive firewall with packet screening rules in place)
- Severity = (5 + 2) – (5 + 5) = ***-3***

9. Defensive recommendation:

- Defenses are OK since the Gauntlet firewall blocked the scan.
- Sweep all addresses in the DMZ for machines with port 137 open and close NetBIOS services on port 137-139 if found (a bit anal maybe, but it's OK to be too careful here).
- Apply current patches to firewall (if current patch set is newer).
- Add access control lists on the screening router to deny connection attempts to port 137 from outside the DMZ (another layer of defense).

10. Multiple choice question based on trace:

- What type of service was targeted in this scan?
  F. NetBIOS name service
  G. NetBIOS datagram service
  H. NetBIOS session service
  I. NetBEUI
  Answer: A

Detect #3

1. Source of trace:

- http://www.sans.org/y2k/013000-1200.htm
- This quote is from the link above: "Here is an interesting one, I let it go a bit long, for my students who are looking for their analyst's certifications, and this is a great one to do an analysis on (hint hint). We see an Unix-like looking start and then an apparently random port scan. Question to the Trojan hounds, is this just a port scan or could these be targets. Note: there is lots of good stuff after these scans so stay tuned!"

```
Jan 26 00:51:05 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 79
Jan 26 00:53:51 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 111
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 363
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 236
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 673
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 1007
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 317
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 418
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 378
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 29
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 363
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 979
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
```

scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 207
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 980
Jan 26 01:20:29 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 532
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 481
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 517
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 595
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 508
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 586
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 260
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 727
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 248
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 117
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 163
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 331
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 411
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 446
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 541
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 726
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 202
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 263
Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal
scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 250

2. Detect was generated by:

- The URL listed above did not specify the exact source of this trace, however it is clear that this trace came from the syslog on a machine running some flavor of UNIX.
- Portsentry. Portsentry is a 3rd party tool for detecting port scans in real-time, and it records its alerts to the system's syslog. This trace originates from the syslog on a UNIX box named "cybernet".
- Log format for portsentry within syslog:
  - Jan 26 01:20:30 cybernet portsentry[18767]: attackalert: SYN/Normal scan from host: adsl-77-244-119.mia.bellsouth.net/216.77.244.119 to TCP port: 250
  - <Timestamp> <loghostname> <portsentry's PID> <type of scan> <source address> <protocol> <port number>

- The relevant information for this trace follows:
  - Type of scan: SYN
  - Source address: 216.77.244.119
  - Destination: host "cybernet"
  - Protocol: TCP
  - Destination ports: [29 – 1007] including some well-known services such as finger and sunrpc.

3. Probability the source address was spoofed (probably spoofed, probably not spoofed, or 3rd party):

- Very low. This is a recon probe (a randomized port scan instead of a sequential one in this case) of a host for services and possibly Trojan ports. Port 79 is well known as finger, but it is also a Trojan port for CDK and Firehotcker. The attacker would need the information returned to a valid address for information gathering purposes, so the source address is probably not spoofed.

4. Description of attack:

- This attack is a randomized port scan on a host looking for well-known services and possibly Trojan ports. A longer network trace would have confirmed or denied this, but the fact that port 79 is also a Trojan port raises that possibility.
- The attacker could use the information from this probe to exploit vulnerabilities on well-known services such as finger and sunrpc.
- The attacker could also use the information from this probe to find backdoors on host "cybernet" if a Trojan was discovered.

5. Attack mechanism:

- This attack is a stimulus designed to find ports that are listening for incoming connections.
- A randomized list of services is being targeted. Some of these services are finger, sunrpc, msg-icp, and Apple-Talk Name Binding.
- Of the services scanned above, sunrpc is one of the most vulnerable as evidenced by the following CVEs:
  - CVE-1999-0696 – rpc.cmsd
  - CVE-1999-0018, CVE-1999-0019 – rpc.statd. A buffer overflow in rpc.statd can give an attacker root privileges.
- This attack sends a SYN packet to each port and waits for a response. If a SYN/ACK is returned, then the port is listening. A RST packet indicates that the port is not listening. Nmap is capable of such a randomized port scan using SYN packets.

6. Correlations:

- This detect was credited to SANS GIAC on January 30, 2000.
- A whois query of the subnet address "216.77.244" indicates that BellSouth.net owns the following network address block: 216.76.0.0 - 216.79.255.255
- A search of the subnet address "216.77." SANS GIAC's reveals the following activity from BellSouth.net:
  - http://www.sans.org/y2k/041100.htm - Trojan scan for SubSeven, Netbus, Netbus 2, and Sub-7 2.1
  - http://www.sans.org/y2k/061600.htm – Trojan scan for SubSeven and Blade Runner and an attempt to ftp port 21.
  - http://www.sans.org/y2k/083000-1430.htm - Scans against a few targeted hosts and DNS queries.

- A search for activity from BellSouth.net against FIWC's CND database reveals Trojan probe activity on at least 17 occasions.
- Traffic from BellSouth.net indicates that users are scanning for open well-known services and open Trojan ports against both commercial and military targets.

7. Evidence of active targeting:

- Active targeting is evident by the scanning of a single host for open ports. These ports are among the well-known services, however port 79 is also listed at these URLs as a Trojan port:
  http://www.simovits.com/nyheter9902.html,
  http://www.silverdragon.dyndns.org/trojans/
  http://www.doshelp.com/trojanports.htm
- Port 79 is commonly known as "finger". However, these Trojans also use it: CDK and Firehotcker.

8. Severity:

- (System criticality + Lethality)
  - (System Countermeasures + Network Countermeasures) = Severity
- Target hostname is "cybernet". A little bit of web surfing revealed websites named cybernet.net and cybernet.com. I'll assume that the host "cybernet" is that organization's DNS or email server since the trace above was rather vague.
- System criticality = 5 (DNS server)
- Lethality = 2 (recon scan)
- System Countermeasures = 5 (another assumption but the sys admin was diligent enough to supplement his/her syslog with the 3rd party logging tool portsentry, so I'll assume his system is patched as well)
- Network Countermeasures = 2 (assuming a firewall is in place but with no indication of its restrictive measures)
- Severity = (5 + 2) – (5 + 2) = ***0***

9. Defensive recommendation:

- Install a firewall and screening router and configure them with the appropriate packet screening rules and access control lists (for the screening router) to deny this type of traffic.
- Install an intrusion detection system to supplement the logging features already in place.
- Check the host "cybernet" for more open ports to ensure that no unneeded services are running or that no Trojan ports are open.
- It's unclear if a Trojan is running on port 79, but a thorough check of host "cybernet" is strongly recommended.
- Traffic from BellSouth.Net should be closely monitored given the amount of this type of activity towards both commercial and military targets.

10. Multiple choice question based on trace:

- What Trojan uses port 79?
  A. finger
  B. SubSeven
  C. BackOrifice 2K
  D. Firehotcker
  Answer: D

Detect #4

1. Source of trace:

- I decided to poke around on our firewall for some interesting activity and found attempted DNS zone transfers from Korea. I searched SANS GIAC and found similar activity originating from another network belonging to the Korea Network Information Center. Attempted zone transfers and rpc dumps from unfriendly source IP addresses should be closely monitored. This isn't an entirely exciting set of traces, but it's interesting to note that the activity came from an unfriendly source IP address and was directed to a commercial site, a university, *and* a military command. Additionally, the traffic followed a pattern of network mapping using a zone transfer followed by an attempt to find open rpc services.
- A poster, from the University of Alabama, to the newsgroup, comp-protocols-dns-bind, also reported similar activity to his network.
- http://www.sans.org/y2k/012100.htm
  Jan 19 03:05:37 point named[130]: unapproved AXFR from [210.218.252.150].1599 for "domain1.ca" (acl)
  Jan 19 03:06:17 point named[130]: unapproved AXFR from [210.218.252.150].1621 for "domain2.ca" (acl)
  Jan 19 03:07:06 point named[130]: unapproved AXFR from [210.218.252.150].1644 for "domain3.ca" (acl)
  Jan 19 03:08:17 point named[130]: unapproved AXFR from [210.218.252.150].1724 for "domain4.gc.ca" (acl)
- http://www.sans.org/y2k/012300.htm
  Jan 21 17:37:43 ns3 rpcbind: refused connect from 210.206.183.131 to dump()
  Jan 21 18:10:39 ns3 rpcbind: refused connect from 210.206.183.131 to dump()
- Fleet Information Warfare Center firewall log
  Oct 1 14:19:28 cartman named[66]: unapproved AXFR from [147.46.119.195].4184 for "FIWC.navy.mil" (acl)
  Oct 1 23:09:59 cartman named[66]: unapproved AXFR from [147.46.119.195].4126 for "FIWC.navy.mil" (acl)
- Fleet Information Warfare Center NetRanger log
  4,2990522,2000/10/01,17:44:45,2000/10/01,17:44:45,10008,99,9999,OUT,OUT,4,6052,0,TCP/IP, 147.46.119.195,x.y.z.2,4184,53,0.0.0.0,
  4,1125521,2000/10/02,02:35:14,2000/10/02,02:35:14,10008,99,9999,OUT,OUT,4,6052,0,TCP/IP, 147.46.119.195,x.y.z.2,4126,53,0.0.0.0,

2. Detect was generated by:

- Syslog
  - Jan 19 03:08:17 point named[130]: unapproved AXFR from [210.218.252.150].1724 for "domain4.gc.ca" (acl)
  - Oct 1 23:09:59 cartman named[66]: unapproved AXFR from [147.46.119.195].4126 for "FIWC.navy.mil" (acl)
  - <Timestamp> <loghostname> <process[process_id]> <process warnings and messages>
    - Timestamp – Jan 19 at 03:08 and Oct 1 at 23:09
    - Log hostname – point and cartman
    - Process[PID] – named with a PID of 130 and 66 respectively
    - Process warnings and messages – An unapproved zone transfer was attempted from 210.218.252.150 and 147.46.119.195 for the domains domain4.gc.ca and fiwc.navy.mil. The access control list for named on both servers did *not* allow the transfer.

0

- NetRanger
  - 4,2990522,2000/10/01,17:44:45,2000/10/01,17:44:45,10008,99,9999,OUT,OUT,4,6052,0, TCP/IP,147.46.119.195,x.y.z.2,4184,53,0.0.0.0,
  - NetRanger log fields
    - Record Type - 4
    - Record ID - 2990522
    - GMT Datestamp - 2000/10/10 (yyyy/mm/dd)
    - GMT Timestamp - 17:44:45 (hh:mm:ss)
    - Local Datestamp - 2000/10/01 (yyyy/mm/dd)
    - Local Timestamp - 17:44:45 (hh:mm:ss)
    - Application ID - 10008
    - Host ID - 99
    - Organization ID - 9999
    - 0Source Direction - OUT
    - Destination Direction - OUT
    - Alarm Level - 4 (Critical)
    - SigID - 6052 (from the NetRanger documentation, DNS High ZoneTransfer)
    - SubSigID - 0
    - Protocol - TCP/IP
    - Source IP Address - 147.46.119.195
    - Destination IP Address - x.y.z.2
    - Source Port - 4184
    - Destination Port - 53 (DNS)
    - Router IP Address - 0.0.0.0 (NetRanger usually sets this to 0.0.0.0)

3. Probability the source address was spoofed (probably spoofed, probably not spoofed, or 3rd party):

- Very low. The information returned from a DNS zone transfer and an rpc dump would have to be returned to a valid IP address for information gathering purposes.

4. Description of attack:

- The attacker(s) attempted to obtain zone files for 3 sites using the "ls" command while running "nslookup". These zone files would have returned every IP address for every zone that the respective machines provide DNS. This is a quick way to do recon (although not a very quiet way).
- The attacker(s) also tried to obtain a list of RPC services that might be vulnerable to a buffer overflow.

5. Attack mechanism:

- Zone transfer – running nslookup and typing "ls <domain name>" at the prompt is the mechanism used to attempt a zone transfer. TCP is needed to ensure reliable delivery of zone files to an authorized DNS server. A query is sent to the domain name server with the QTYPE=AXFR in the Question Section portion of the DNS header (RFCs 1034 and 1035).
- "rpcinfo –p <host>" could be used to check a host for a list of RPC services running on it. RFCs 1050, 1057, and 1833 explain the inner workings of RPC.

6. Correlations:

- The link below is a thread, dated 23 Aug 2000, discussing recent zone transfers coming from Russia, Korea, Sweden and Canada. The sys admin who started the thread is from the University of Alabama.
- http://www.mail-archive.com/comp-protocols-dns-bind%40isc.org/msg00791.html
- Whois lookups
  - 210.218.252.150, 210.206.183.131 - National Computerization Agency, Korea Network Information Center
  - 147.46.119.195 - Seoul National University
- This DNS activity has been observed since the beginning of the year at SANS GIAC, the University of Alabama, and the Fleet Information Warfare Center.
- http://www.sans.org/y2k/012100.htm – Attempted DNS zone transfer
- http://www.sans.org/y2k/012300.htm – Attempted rpc dump
- http://www.sans.org/y2k/020500.htm – Similar pattern of activity. An attempted rpc dump followed an attempted DNS zone transfer.
- No RPC activity has been observed here from the Korean IP addresses, although that is traffic that will be monitored closely.

7. Evidence of active targeting:

- Active targeting is evident since the attacker was looking for very specific host information (zone files) and RPC services.

8. Severity:

- I'll use our network to calculate the severity for this trace.
- (System criticality + Lethality)
  – (System Countermeasures + Network Countermeasures) = Severity
- Target network included DMZ routers/firewall.
- System criticality = 5 (firewall, DMZ, outer screening router)
- Lethality = 2 (recon scan)
- System countermeasures = 5 (system patched)
- Network countermeasures = 5 (restrictive firewall with packet screening rules and ACLs in place)
- Severity = (5 + 2) – (5 + 5) = ***-3***

9. Defensive recommendation:

- The firewall did not allow the zone transfer so the current defensive posture is OK. Running snort or tcpdump on the firewall would provide more meaningful logging than syslog alone.
- Lock down all unnecessary RPC services within the DMZ.
- Double check named.conf to see a list of servers who are allowed to perform zone transfers. Ensure that those boxes (both local and remote secondary servers) have not been compromised.

10. Multiple choice question based on trace:

- What protocol/port is needed to facilitate a zone transfer?
  E. UDP/53
  F. UDP/111
  G. TCP/53
  H. TCP/25
  Answer: C

Evaluate an Attack – NetBIOS scanning

NetBIOS scanning was chosen given the recent increase in this type of activity at the SANS GIAC and its widespread occurrence among commercial, educational, and military sites. Exploiting unprotected shares on Windows machines is quite common, and a few worms, such as network.vbs, propagates across networks using port 137. Proper defensive measures, such as not allowing external NetBIOS connections, should be taken to prevent this type of activity from occurring.

A NetBIOS scanning tool named nbtscan (http://www.abb.aha.ru/software/nbtscan.html) was used to perform the scan on our local network. Using nbtscan is much faster than running "nbtstat –a <hostname>" from the command line on a Windows NT machine.

The command "nbtscan –v x.y.z.0/24" was run from a RedHat Linux 6.1 machine for this analysis. "nbtscan –v" provides a verbose output and prints the NetBIOS name table for each host that responded during the scan. Selected hosts from that scan are shown below (IP and MAC addresses have been sanitized):

*** BEGIN nbtscan output ***

```
Doing NBT name scan for addresses from x.y.z.0/24

NetBIOS Name Table for Host x.y.z.53:

Name            Service          Type
---------------------------------------
CURLY           <20>             UNIQUE
CURLY           <00>             UNIQUE
NOBODY-NT       <00>              GROUP
NOBODY-NT       <1c>              GROUP
NOBODY-NT       <1e>              GROUP
CURLY           <03>             UNIQUE
ROOT            <03>             UNIQUE

Adapter address: 00-90-27-3f-b8-f3
---------------------------------------

NetBIOS Name Table for Host x.y.z.50:

Incomplete packet, 209 bytes long.
Name            Service          Type
---------------------------------------
CARTMAN         <20>             UNIQUE
CARTMAN         <00>             UNIQUE
WORKGROUP       <00>              GROUP
WORKGROUP       <1e>              GROUP
  __MSBROWSE__  <01>              GROUP
                <00>             UNIQUE

Adapter address: cc-25-0f-32-35-00
---------------------------------------
```

```
NetBIOS Name Table for Host x.y.z.54:

Name             Service          Type
---------------------------------------
MOE            <00>             UNIQUE
MOE            <20>             UNIQUE
NOBODY-NT      <00>              GROUP
NOBODY-NT      <1c>              GROUP
FRED           <00>              GROUP
ETHEL          <00>              GROUP
NOBODY-NT      <1b>             UNIQUE
NOBODY-NT      <1e>              GROUP
MOE            <03>             UNIQUE
NOBODY-NT      <1d>             UNIQUE
   __MSBROWSE__ <01>              GROUP
INet~Services  <1c>              GROUP
IS~MOE         <00>             UNIQUE
MOE            <6a>             UNIQUE
MOE            <87>             UNIQUE
MOE            <01>             UNIQUE

Adapter address: 00-90-27-3a-a8-83
```

*** END nbtscan output ***

The NetBIOS Name Table (RFC 1001/1002)
This URL explains the NetBIOS Name Table as used in Windows NT: http://support.microsoft.com/support/kb/articles/Q163/4/09.ASP. The following excerpt from this link explains the service codes from the output above:

Name	Number(h)	Type	Usage
<computername>	00	U	Workstation Service
<computername>	01	U	Messenger Service
<\\--__MSBROWSE__>	01	G	Master Browser
<computername>	03	U	Messenger Service
<computername>	06	U	RAS Server Service
<computername>	1F	U	NetDDE Service
<computername>	20	U	File Server Service
<computername>	21	U	RAS Client Service
<computername>	22	U	Microsoft Exchange Interchange(MSMail Connector)
<computername>	23	U	Microsoft Exchange Store
<computername>	24	U	Microsoft Exchange Directory
<computername>	30	U	Modem Sharing Server Service
<computername>	31	U	Modem Sharing Client Service
<computername>	43	U	SMS Clients Remote Control
<computername>	44	U	SMS Administrators Remote Control Tool
<computername>	45	U	SMS Clients Remote Chat
<computername>	46	U	SMS Clients Remote Transfer
<computername>	4C	U	DEC Pathworks TCPIP service on Windows NT
<computername>	42	U	mccaffee anti-virus
<computername>	52	U	DEC Pathworks TCPIP service on Windows NT

<computername>	87	U	Microsoft Exchange MTA
<computername>	6A	U	Microsoft Exchange IMC
<computername>	BE	U	Network Monitor Agent
<computername>	BF	U	Network Monitor Application
<username>	03	U	Messenger Service
<domain>	00	G	Domain Name
<domain>	1B	U	Domain Master Browser
<domain>	1C	G	Domain Controllers
<domain>	1D	U	Master Browser
<domain>	1E	G	Browser Service Elections
<INet~Services>	1C	G	IIS
<IS~computer name>	00	U	IIS
<computername>	[2B]	U	Lotus Notes Server Service
IRISMULTICAST	[2F]	G	Lotus Notes
IRISNAMESERVER	[33]	G	Lotus Notes
Forte_$ND800ZA	[20]	U	DCA IrmaLan Gateway Server Service

As shown above, it is clear to see why NetBIOS scanning can be a key component in network reconnaissance. It is interesting to note that the information returned from this scan is somewhat similar to the output returned from “rpcinfo –p <hostname>” on a UNIX box.

Snort Log excerpt
The following Snort v1.6.3 trace contains the respective hosts from the scan above.

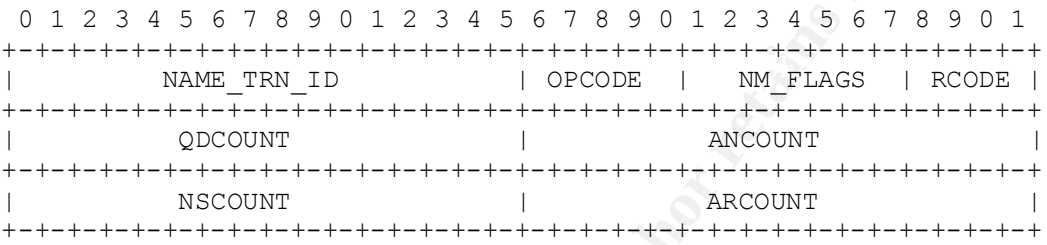
```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
10/10-17:29:40.182578 x.y.z.77:1842 -> a.b.c.50:137 UDP TTL:64
TOS:0x0 ID:29658
Len: 58
00 B6 00 10 00 01 00 00 00 00 00 00 20 43 4B 41  ............ CKA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 00 00 21  AAAAAAAAAAAAA..!
00 01                                            ..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
10/10-17:29:40.182784 x.y.z.77:1842 -> a.b.c.53:137 UDP TTL:64
TOS:0x0 ID:29661
Len: 58
00 B6 00 10 00 01 00 00 00 00 00 00 20 43 4B 41  ............ CKA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 00 00 21  AAAAAAAAAAAAA..!
00 01                                            ..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
10/10-17:29:40.182876 x.y.z.77:1842 -> a.b.c.54:137 UDP TTL:64
TOS:0x0 ID:29662
Len: 58
00 B6 00 10 00 01 00 00 00 00 00 00 20 43 4B 41  ............ CKA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 00 00 21  AAAAAAAAAAAAA..!
00 01                                            ..
```

Snort log format

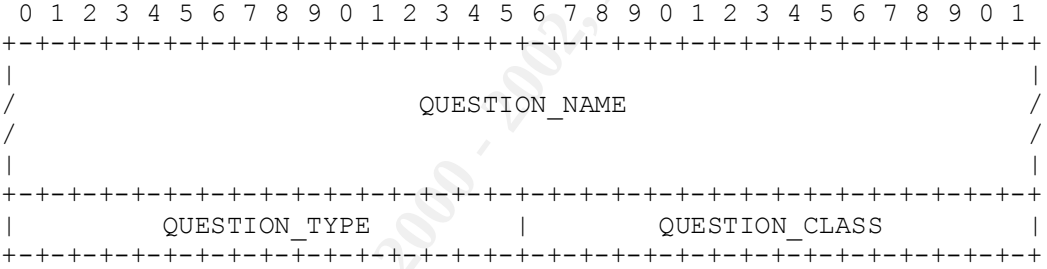
{Date} {hh:mm:ss.fraction_of_a_second} {Source IP address} -> (direction of data flow) {Destination IP address} Protocol_header_info (ICMP, TCP, UDP. UDP in this trace) flags ID

Data in hexadecimal format Data in ASCII format
10/10-17:29:40.182876 x.y.z.77:1842 -> x.y.z.54:137 UDP TTL:64 TOS:0x0 ID:29662
Date – 10/10
Time – 17:29:40.182876
Source – x.y.z.77, port 1842
Destination – a.b.c.54, port 137
Protocol – UDP
TTL – Time To Live
TOS – Type of Service
ID - ID

NetBIOS Header (from RFC 1002, section 4.2.1.1)



NetBIOS Question section (from RFC 1002, section 4.2.1.2)



Explanation of snort trace by byte order (byte(s) – name of field – value in hex)

- 0 & 1 – Name transaction ID – 00 B6
- 2 & 3 – Opcode, NM_flags, & Rcode – 00 10
  - 00 – request, query packet
  - 10 – broadcast/multicast packet
- 4 & 5 – QDcount – number of entries in the question section of a name – 00 01
  - 00 01 – 1 entry or query
- 6 – 11 - ANCount, NSCount, ARCount – not used – 00 00 00 00 00 00
- 12 – Size of QUESTION_NAME field - 20
- 13 – 45 – QUESTION_NAME field
  - The "CKAAA…A" value represents the query sent to each machine in the scan. The "CKAAA…A" string actually represents an asterisk "*" followed by nulls. An asterisk will prompt the remote machine to send the contents of its NetBIOS name table.
  - From RFC 1002
    NODE STATUS REQUEST:

```
/*
 * Name of "*" may be used for force node to
 * divulge status for administrative purposes
 */
```

- 46 – Null field – 00
- 47 & 48 – QUESTION_TYPE field, type of request – 00 21

    - 00 21 – Node status request
- 49 & 50 QUESTION_CLASS field, class of the request – 00 01
    - 00 01 – Internet class

The attack works by sending a "NODE STATUS REQUEST" to the remote machine. The remote machine will send its name table if the value of the name it receives is a "*". This name table will reveal available services, machine names, Windows NT workgroup names, and usernames (where available).

This very simple Snort rule was used to detect the scan above: log udp any any -> a.b.c.0/24 137 (no specific signature matches were used for this trace). This rule was used for the analysis since many of the NetBIOS scans witnessed here and at the SANS GIAC involve a destination port of 137 (and not 138 or 139). Another Snort rule using the "CKAAA…AAA" string above could also be used in a Snort rule to capture NetBIOS traffic. Snort rules incorporating ports 138 and 139 should also be used for completeness to help detect NetBIOS activity.

Analyze This – Attack on MY.NET (known hereafter as 10.10)

Scenario:
A facility has asked us to provide security services for them and has allowed us to run Snort for a month using a fairly standard rulebase. We experienced power failures and disk problems during this time, so the data is incomplete. Our task is to analyze the data for signs of compromised systems or network problems.

Report:
Machines on the 10.10 net Compromised:

I assumed immediately that one or more machines on the network would have been compromised before our arrival. I checked the Snort logs for evidence of this and it is highly probable that the following machines are compromised:

- 10.10.1.3
- 10.10.97.237
- 10.10.101.192

Common ports

- 161 – SNMP
- 8080 – WinGate
- 27374 – Sub7
- 53 – DNS
- 34555 – Trinoo
- 137 – NetBIOS

I recommend that you immediately disconnect these machines from the network. These are being used to scan other machines on the network. 10.10.101.89 bears the brunt of this attack since these 3 machines are heavily port scanning (sequentially) it. 10.10.97.237 was scanned on July 9 for port 44767 and on July 11 for port 27374 (Sub7). There is no currently known Trojan for port 44767, but the following 2 sites have reported activity on this port.

http://archives.neohapsis.com/archives/incidents/2000-05
http://www.sans.org/y2k/052400.htm

Jul 9 08:13:37 212.29.71.87:3724 -> 10.10.97.237:44767 UDP
Jul 9 09:29:53 213.14.3.102:4373 -> 10.10.97.237:44767 UDP
Jul 11 18:48:07 4.54.218.182:3928 -> 10.10.97.237:27374 SYN **S*****
Jul 11 18:48:09 4.54.218.182:3928 -> 10.10.97.237:27374 SYN **S*****

Starting on July 14 10.10.97.237 began querying 10.10.101.192 for SNMP public access. It's probable that a remote-access Trojan like Sub7 exists on 10.10.97.237 allowing outside entity access into the private network.

Suspicious traffic (DNS scanning) was also observed from 211.60.222.33. Activity from this network has also been observed at the SANS GIAC at this URL: http://www.sans.org/y2k/012100.htm.
This IP address is owned by

inetnum	211.52.0.0 - 211.63.255.255
netname	KRNIC-KR-24
descr	Korea Network Information Center
descr	14F, NARA Bldg, 1328-3, Seocho-Dong, Seocho-Ku
descr	Seoul, Korea, 137-070
country	KR
admin-c	WK1-AP, inverse
tech-c	SL119-AP, inverse
remarks	KRNIC Allocation Block
remarks	Authoritative Information regarding assignments and
remarks	allocations made from within this block can also be
remarks	queried at whois.nic.or.kr
mnt-by	APNIC-HM, inverse
mnt-lower	MNT-KRNIC-AP, inverse
changed	hostmaster@apnic.net 20000216
source	APNIC

This traffic was scanning for DNS servers on your network. Over a 5-hour period, DNS on your network was scanned over 23000 times.

Suspicious traffic between port 25 (SMTP) and 34555 (Trin00)
Another disturbing trend was discovered during our short stint. There was activity between ports 25 and 34555 beginning on June 27 and ending on August 1.

```
awk -F\  '{print $1,$9,$11}' 34555 | sort | uniq -u | more
06/27-03:54:29.006819 192.101.175.131:25 10.10.100.230:34555
06/27-09:44:44.393993 207.172.4.98:25 10.10.253.52:34555
06/27-20:53:42.310590 216.33.151.135:25 10.10.253.53:34555
06/28-07:35:07.079259 165.251.8.33:25 10.10.253.24:34555
06/28-17:48:52.956314 208.128.229.254:25 10.10.100.230:34555
06/28-21:24:55.303105 24.0.95.22:25 10.10.253.51:34555
06/29-14:15:40.983712 131.96.1.22:25 10.10.253.24:34555
06/30-14:55:26.571163 140.142.100.15:25 10.10.253.52:34555
skipping lines here
07/30-07:32:40.110925 152.163.224.100:25 10.10.253.24:34555
07/30-21:43:50.908928 143.207.1.8:113 10.10.6.34:34555
08/01-01:22:54.041442 24.2.2.66:113 10.10.6.47:34555
08/01-06:03:56.320188 169.226.1.24:113 10.10.253.42:34555
08/01-10:24:53.495542 209.27.89.5:25 10.10.253.51:34555
```

```
192.101.175.131
MRJ INC. (NET-MRJ)
10560 Arrowhead Drive
Fairfax, VA 22030

Netname: MRJ
Netnumber: 192.101.175.0

Coordinator:
Hern, Tony (TH4156-ARIN) tony.hern@MRJ.COM
(703)277-1215 (FAX) (703)385-4637

Domain System inverse mapping provided by:
```

NS2.MRJ.COM 192.101.175.2
DNS2-M.ANS.NET 198.83.47.41

207.172.4.98
Erol's Internet Services (NETBLK-NETBLK-EROLS-BLK-3)
7921 Woodruff Ct.

Springfield, VA 22151

Netname: NETBLK-EROLS-BLK-3

Netblock: 207.172.0.0 - 207.172.255.255

Maintainer: EROL

Coordinator:

Network Operations Center (EROLS-NOC-ARIN) noc@RCN.COM

703-321-8000

Fax- 703-321-8316

Domain System inverse mapping provided by:

AUTH1.DNS.RCN.NET 207.172.3.20

AUTH2.DNS.RCN.NET 206.138.112.20

AUTH3.DNS.RCN.NET 207.172.3.21

AUTH4.DNS.RCN.NET 207.172.3.22

24.2.2.66
@Home Network (NETBLK-MD-RDC-1)
425 Broadway
Redwood City, CA 94063 US

Netname: MD-RDC-1
Netblock: 24.2.2.0 - 24.2.2.255

Coordinator:
Operations, Network (HOME-NOC-ARIN) noc@NOC.HOME.NETabuse@corp.home.net
1-800-872-3595

These machines should be scanned for Trojans.

WinGate
This is a Windows firewall that is frequently misconfigured and is popular method for hackers to tunnel and forward their attacks. The following CVEs at cve.mitre.org explains these vulnerabilities: CVE-1999-0290, CVE-1999-0291, CVE-1999-0441, and CVE-1999-0494. These machines saw the most activity:

- 10.10.60.11
- 10.10.60.8
- 10.10.60.16

These machines should be reconfigured and hardened.

SYNFIN

At least 843 SYN/FIN scans from source port 53 to destination port 53 occurred during our stint. Both the SYN and the FIN flags are set in TCP packets sent to the destination hosts. Firewalls will let port 53 traffic through and this method can be used to circumvent them. This type of traffic is abnormal and is a common method to map networks and identify target machine operating systems.

Some whois lookups on machines who performed SYN/FIN scans.
210.84.179.196
netname OZEMAIL2-AU
descr OzEmail Pty Ltd
country: AU

193.173.174.119
netname: SCARAMEA
descr: e-commerce
descr: internet service provider
country: NL

211.7.235.4
netname JPNIC-NET-JP
descr Japan Network Information Center
country JP

Summary
Several foreign countries, including Korea, the Netherlands, Japan, and Australia have actively targeted your network. These countries are well known for their research in computer security and information warfare. At least 3 of your machines have been compromised, and one of them is scanning the rest of your network for vulnerabilities. You should take the appropriate defensive measures to lockdown your network: disconnecting compromised machines, scanning them for Trojans, beefing network defenses through more stringent packet filtering rules, router access control lists, improved logging software, and commercial intrusion detection systems, user education and training.

Analysis Process

- There were 37 "alerts" and "scan" files at http://www.sans.org/PH2000/snort/index.htm I concatenated (UNIX "cat" command) the "alerts" (SnortA*) files and the "scan" files (SnortS*) into 2 files: "sn-alerts.txt" and "sn-scans.txt" (known hereafter as "alerts" and "scans") respectively.
- Tools used
  - SnortSnarf – I used the Perl script, snortsnarf.pl, to parse "alerts" and "scans". This tool parses Snort alerts and produces HTML files for diagnosing problems. Screenshots for each type of trace are shown below:
  - Grep – I used this simple UNIX command to find a list of machines on the 10.10 network that may be the source of illegal activity. Once these machines were found, I grepped for these machines as the *destination* address for activity from external IP addresses. These approaches quickly aided in finding machines that were compromised from external IP addresses.
  - Snort-sort – I used this script to produce a list of sorted alarms. This sorted list and the output from the "grep" commands helped to prioritize our analysis and quickly determine a set of machines that were probably compromised.

Figure 1 - SnortSnarf for sn-alerts.txt

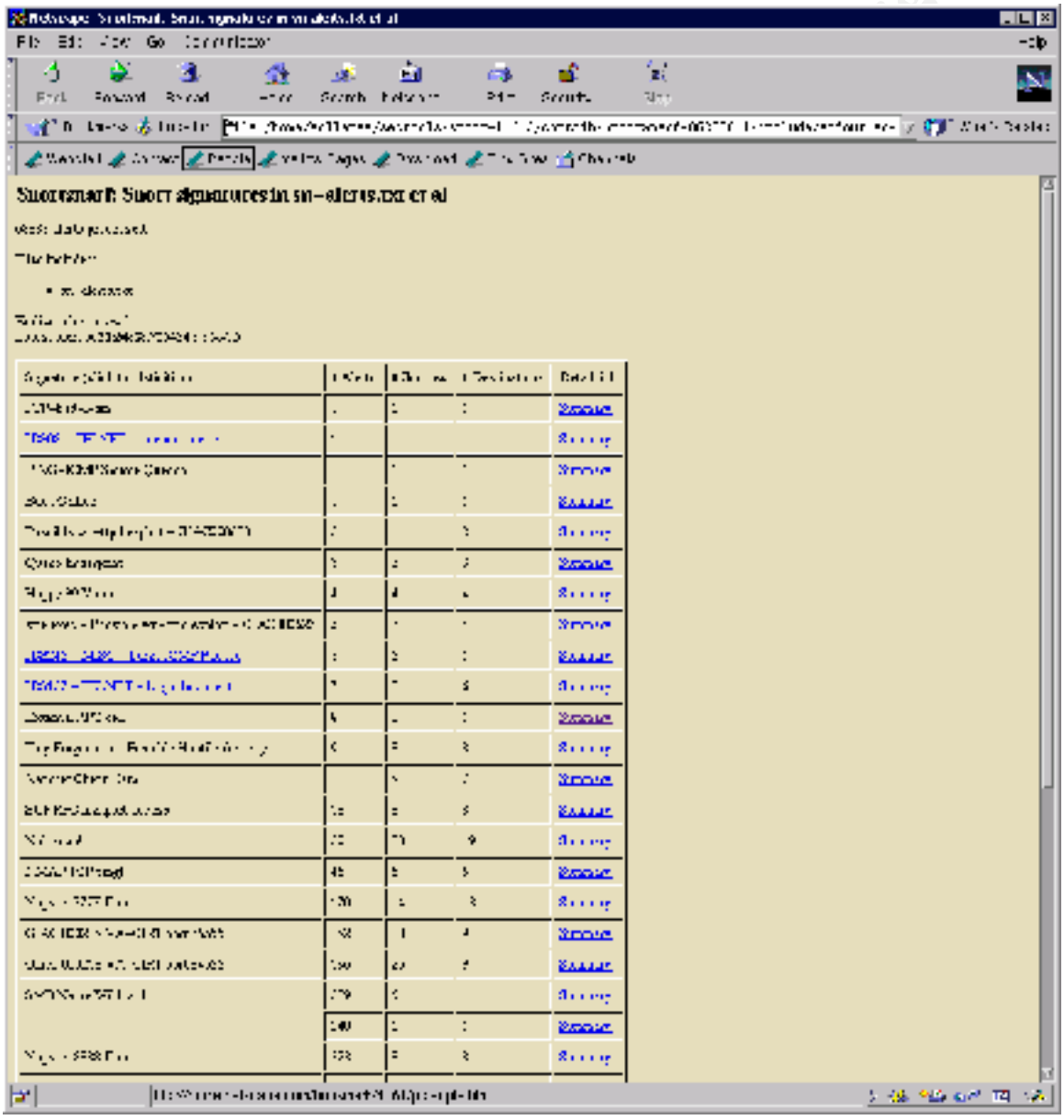
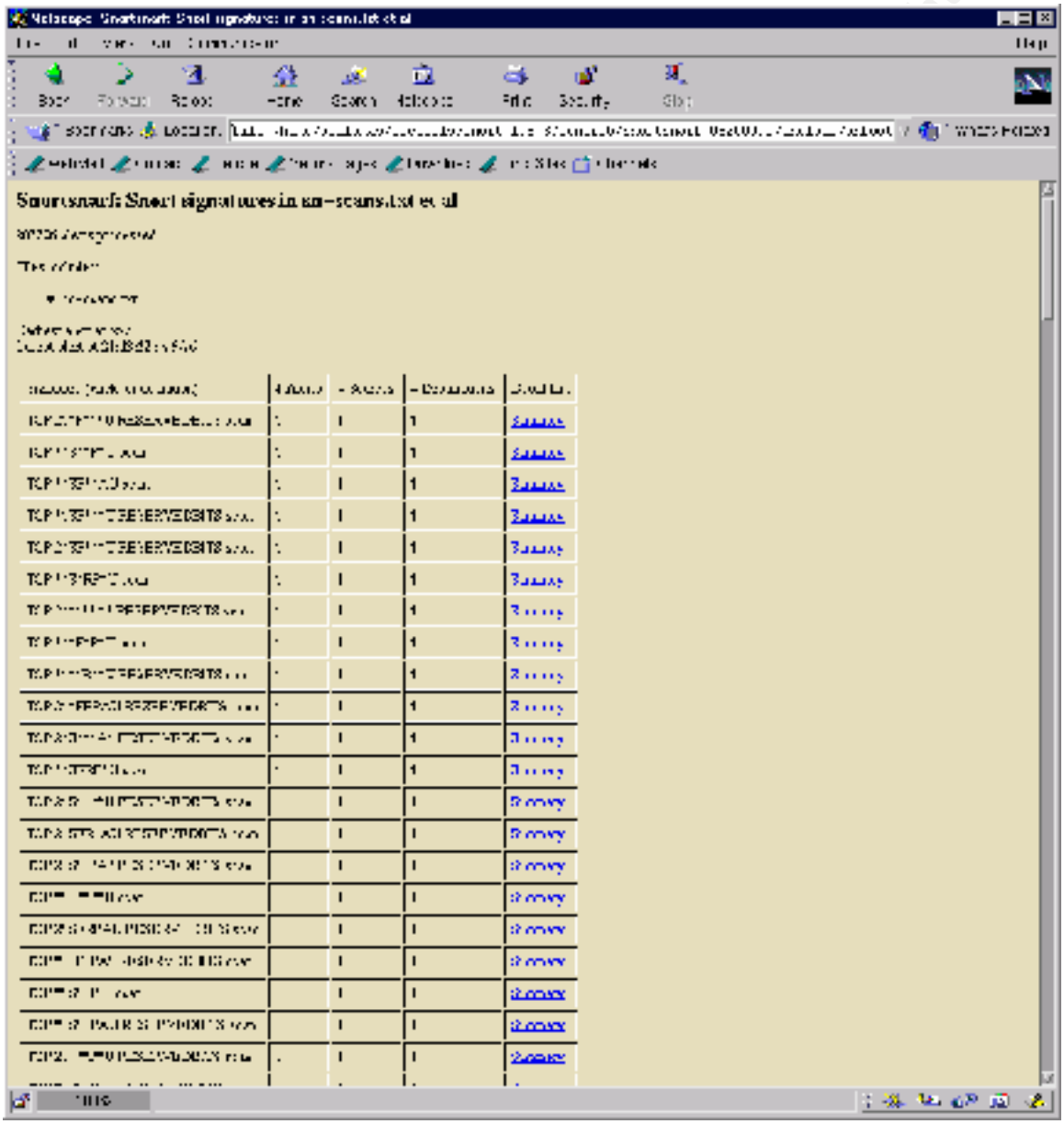


Figure 2 - SnortSnarf for sn-scans.txt



- Grep – I used the following grep commands to produce a listing of machines on the 10.10 network, which may be the sources of alarms and scans (I assumed immediately that one or more machines would be compromised).

Figure 4 - grep "10.10..*:.* -> " sn-alerts.txt > alarms.txt

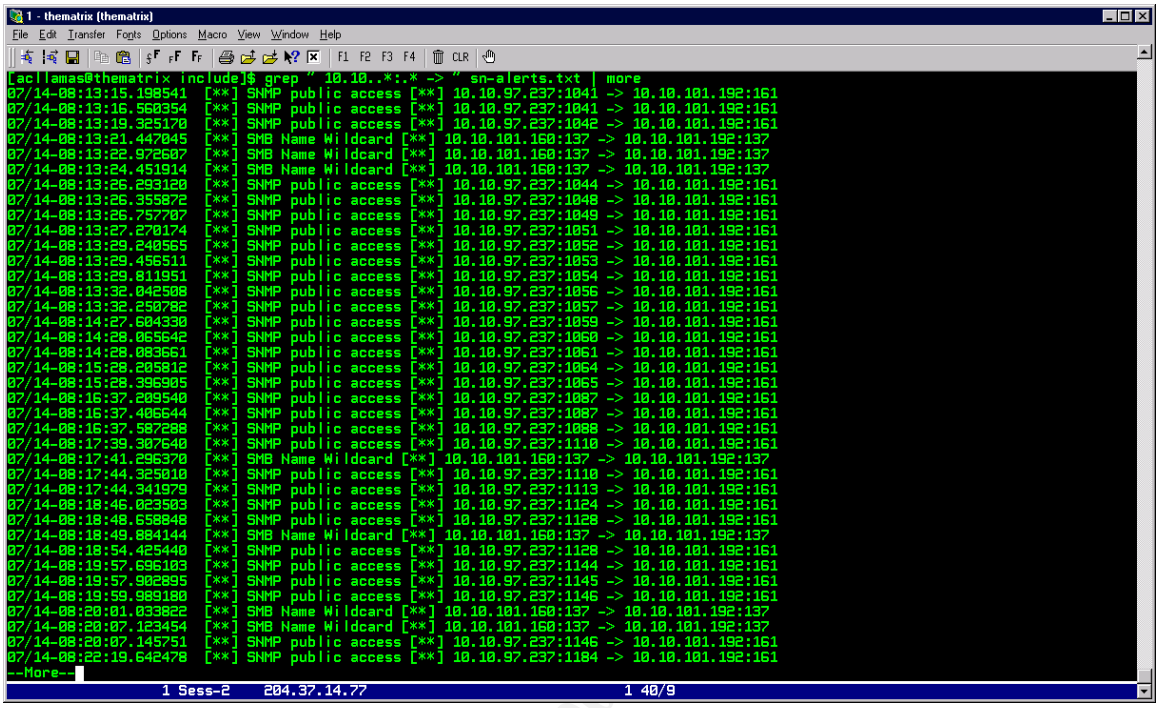


Figure 3 - grep "10.10..*:.* -> " sn-scans.txt > compromised.txt

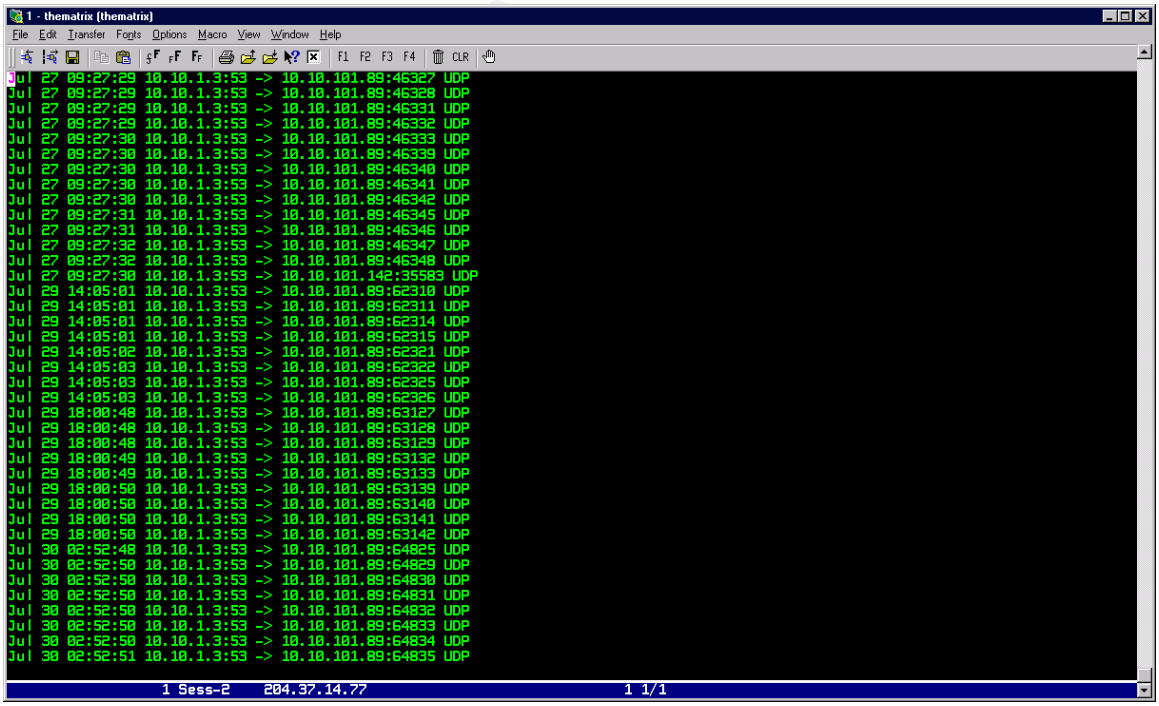


Figure 5 - The Perl script "Snort-sort" was used to produce as sorted list of alarms.