Global Information Assurance Certification Paper

*** Northcutt, fun read. It is great to use an analysis process, but you probably mean criticality of the target! Shadow traces, ummm a fine choice, points for initiative :) In trace 2 if the address is spoofed how can you get information from the probe? Trace 3 you know an address is spoofed, how? Because the IP IDs are in a sequence? Good use of the -v, but splain it to us Lucy! Trace 6, could the options have a story to tell? Watch for this guy to become a strong analyst! 83 ***

Packet Traces with Analysis

This document represents the practical portion of the certification exam given during the IDIC at SANS 2000 in sunny Orlando.

This exam is respectfully submitted by:

Jim Kirby

on

Friday, April 07, 2000

Throughout this document, assessed severity of the attacks will be calculated using the formula given in the IDIC curriculum. Specifically,

S=(C+L)-(SCM+NCM)

Where:

S = Severity of the attack
C = Criticality of the attack
L = (Potential) Lethality of a successful attack
SCM = System Countermeasure Rating
NCM = Network Countermeasure Rating

And each value ranges from 1 to 5. NCM always gets 5 in these examples, however, since it is a packet filter and blocked all listed attacks.

All traces come from a hastily configured SHADOW system, except where indicated.

Trace #1

```
15:46:46.952263 OurNet2.12 > OurNet.10: icmp: echo request
15:46:46.953314 OurNet.10 > OurNet2.12: icmp: echo reply
15:46:47.947321 OurNet2.12 > OurNet.10: icmp: echo request
15:46:47.953372 OurNet.10 > OurNet2.12: icmp: echo reply
15:46:49.377347 OurNet2.12 > OurNet.12: icmp: echo request
15:46:49.377801 OurNet.12 > OurNet2.12: icmp: echo reply
15:46:50.368468 OurNet2.12 > OurNet.12: icmp: echo request
15:46:50.368906 OurNet.12 > OurNet2.12: icmp: echo reply
15:47:05.146407 OurNet2.12 > OurNet.16: icmp: echo request
15:47:05.146752 OurNet.16 > OurNet2.12: icmp: echo reply
15:47:06.135961 OurNet2.12 > OurNet.16: icmp: echo request
15:47:06.136283 OurNet.16 > OurNet2.12: icmp: echo reply
15:47:09.941516 OurNet2.12 > OurNet.18: icmp: echo request
15:47:09.941848 OurNet.18 > OurNet2.12: icmp: echo reply
15:47:10.936039 OurNet2.12 > OurNet.18: icmp: echo request
15:47:10.936370 OurNet.18 > OurNet2.12: icmp: echo reply
15:47:12.846607 OurNet2.12 > OurNet.20: icmp: echo request
15:47:12.846946 OurNet.20 > OurNet2.12: icmp: echo reply
15:47:13.835356 OurNet2.12 > OurNet.20: icmp: echo request
15:47:13.835713 OurNet.20 > OurNet2.12: icmp: echo reply
15:47:21.718094 OurNet2.12 > OurNet.22: icmp: echo request
15:47:21.718469 OurNet.22 > OurNet2.12: icmp: echo reply
15:47:22.712667 OurNet2.12 > OurNet.22: icmp: echo request
15:47:22.713035 OurNet.22 > OurNet2.12: icmp: echo reply
15:47:24.175631 OurNet2.12 > OurNet.24: icmp: echo request
15:47:24.176190 OurNet.24 > OurNet2.12: icmp: echo reply
15:47:25.174927 OurNet2.12 > OurNet.24: icmp: echo request
15:47:25.175291 OurNet.24 > OurNet2.12: icmp: echo reply
15:47:27.003652 OurNet2.12 > OurNet.26: icmp: echo request
15:47:27.003997 OurNet.26 > OurNet2.12: icmp: echo reply
15:47:27.991687 OurNet2.12 > OurNet.26: icmp: echo request
```

Existence: Source IP belongs on our failover Internet connection

History: This is the first IDS alert from that largely unused subnet.

Techniques: Basic ping scan.

Intent: Host mapping.

Targeting: Targeting enumerated hosts.

Severity: (2+1) – (2+5) = -4

Analysis: It appears that someone on, or pretending to be on, our backup Internet connection is using ICMP to enumerate hosts on our primary network. A brief talk with the SA's and it turns out they were "just checking to see if we could ping our hosts from the Internet." False positives are the good positives but the ports on the OurNet2 have been disabled. ;)

Trace #2

```
10:47:58.987769 211.49.194.72.100 > OurNet.16.1672: SFP
5242919:5242939(20) ack 686969152 win 20496 urg 36199 (DF) (ttl 115,id
14619)
```

Existence: Source IP belongs to Thrunet Co, LTD. In Seoul, Korea.
History: None. This is the only packet we have detected from this network.
Techniques: All TCP flags set w/ the urgent pointer being extremely high. Definitely anomalous.
Intent: Normally this type of attack is used for port or host scanning, it is hard to discern intent with only on packet on one port over a 5 day period.
Targeting: Definitely targeted to one of our known hosts.
Severity: (2+2) – (5+5) = -6
Analysis: This is the only invalid-flag packet we've seen since implementing SHADOW. Could this be a corrupted packet that somehow made it to us? Most likely not. More likely, the Korean address is spoofed and a visitor to our server may have been digging for information. What, hard to tell since even the destination port is dubious and not of a known Trojan. My guess is someone fat fingered their probe and typed in our address instead of someone else. The firewall successfully blocked this attack so there was no real threat.

Trace #3

```
07:38:52.010523 169.254.94.104.137 > OurNet.2.137: udp 50 (ttl 118, id 46322)
07:38:52.027246 207.172.37.147.137 > OurNet.2.137: udp 50 (ttl 118, id 46578)
07:38:53.498651 169.254.94.104.137 > OurNet.2.137: udp 50 (ttl 118, id 46834)
07:38:53.507509 207.172.37.147.137 > OurNet.2.137: udp 50 (ttl 118, id 47090)
07:38:54.997116 169.254.94.104.137 > OurNet.2.137: udp 50 (ttl 118, id 47346)
07:38:55.027855 207.172.37.147.137 > OurNet.2.137: udp 50 (ttl 118, id 47602)
07:39:02.695115 169.254.94.104.137 > OurNet.3.137: udp 50 (ttl 118, id 48882)
07:39:02.724882 207.172.37.147.137 > OurNet.3.137: udp 50 (ttl 118, id 49138)
07:39:07.455380 169.254.94.104.137 > OurNet.3.137: udp 50 (ttl 118, id 49394)
07:39:07.464581 207.172.37.147.137 > OurNet.3.137: udp 50 (ttl 118, id 49650)
07:39:08.944226 169.254.94.104.137 > OurNet.3.137: udp 50 (ttl 118, id 49906)
07:39:08.974544 207.172.37.147.137 > OurNet.3.137: udp 50 (ttl 118, id 50162)
07:39:36.907120 169.254.94.104.137 > OurNet.5.137: udp 50 (ttl 118, id 54514)
07:39:36.915327 207.172.37.147.137 > OurNet.5.137: udp 50 (ttl 118, id 54770)
07:39:38.404332 169.254.94.104.137 > OurNet.5.137: udp 50 (ttl 118, id 55026)
07:39:38.434785 207.172.37.147.137 > OurNet.5.137: udp 50 (ttl 118, id 55282)
07:39:39.905351 169.254.94.104.137 > OurNet.5.137: udp 50 (ttl 118, id 55538)
```

```
07:39:39.914171 207.172.37.147.137 > OurNet.5.137: udp 50 (ttl 118, id 55794)
…
```

Existence: 2 IP address appearing to come from opposite coasts one being Erols (which I think is East Coast only) and the U of Southern California at Marina Del Rey. Obviously one, if not both, of the address is spoofed.

History: These connect attempts went on to scan each of our public addresses on that subnet.

Techniques: Basic script-kiddie automated netBios scan looking for open NetBios hosts.

Intent: 911/Chode virus looking for hosts? Definitely scanning for open hosts. Possibly attempting DoS on the 2 source addresses.

Targeting: Scary. Targets NOT each address in the domain, NOT each address that is resolvable, NOT each address that responds to pings, but each address that actually carries data. This shows an extreme knowledge of our network.

Severity: (4+4) – (2+5) = 1

Analysis: This scan falls right in with the 137-137 being reported on GIAC these past days. At least one of these IP addresses is spoofed. This is evident not only from the identical TTLs but the packet ID's and time stamps indicate that the attacker activated 2 scripts simultaneously (or 2 threads in the same attack program) with each instance sending 3 crafted UDP packets at each host. While this does not fit the 5 packets reported at GIAC, it is perhaps a different signature of the same worm.

Trace #4

```
19:49:19.651572 dsl-208-161-106-59.easystreet.com.2952 > OurNet.26.sunrpc: tcp 0 (DF)
19:49:22.450472 dsl-208-161-106-59.easystreet.com.2928 > OurNet.2.sunrpc: tcp 0 (DF)
19:49:22.450600 OurNet.2.sunrpc > dsl-208-161-106 59.easystreet.com.2928: tcp 0
19:49:22.452408 dsl-208-161-106-59.easystreet.com.2929 > OurNet.3.sunrpc: tcp 0 (DF)
19:49:22.465185 dsl-208-161-106-59.easystreet.com.2931 > OurNet.5.sunrpc: tcp 0 (DF)
19:49:22.489098 dsl-208-161-106-59.easystreet.com.2936 > OurNet.10.sunrpc: tcp 0 (DF)
19:49:22.498287 dsl-208-161-106-59.easystreet.com.2938 > OurNet.12.sunrpc: tcp 0 (DF)
19:49:22.517417 dsl-208-161-106-59.easystreet.com.2942 > OurNet.16.sunrpc: tcp 0 (DF)
19:49:22.526931 dsl-208-161-106-59.easystreet.com.2944 > OurNet.18.sunrpc: tcp 0 (DF)
19:49:22.537403 dsl-208-161-106-59.easystreet.com.2946 > OurNet.20.sunrpc: tcp 0 (DF)
19:49:22.547367 dsl-208-161-106-59.easystreet.com.2948 > OurNet.22.sunrpc: tcp 0 (DF)
19:49:22.560435 dsl-208-161-106-59.easystreet.com.2950 > OurNet.24.sunrpc: tcp 0 (DF)
19:49:22.566750 dsl-208-161-106-59.easystreet.com.2952 > OurNet.26.sunrpc: tcp 0 (DF)
```

Existence: Exists and belongs to Cable & Wireless, probably DSL pool.

History: Again, no previous history with these addresses.

Techniques: Scripted RPC host scan.

Intent: Appears to be scanning for SunRPC ports on our network. RPC is a well known vulnerability

Targeting: Another well very targeted attack. Again attacks only those addresses that actually carry data, which is a different set than what could be enumerated via ICMP or DNS lookups. Scary stuff, that.

Severity: (3+3) – (3+5) = -2

Analysis: Someone scanning for open RPC ports on our network. Scary thing is, this coupled with the previous detect is leading me to believe someone along our path to the internet is sniffing our traffic.

Trace #5

```
09:54:37.629484 1.2.2.109.137 > OurNet.16.137: udp 50 (ttl 113, id
5827)
                         4500 004e 16c3 0000 7111 4bdf 0102 026d
                         cc7e 1710 0089 0089 003a fd58 dbd8 0000
                         0001 0000 0000 0000 2043 4b41 4141 4141
                         4141 4141 4141
09:54:39.108520 1.2.2.109.137 > OurNet.16.137: udp 50 (ttl 113, id
6339)
                         4500 004e 18c3 0000 7111 49df 0102 026d
                         cc7e 1710 0089 0089 003a fd56 dbda 0000
                         0001 0000 0000 0000 2043 4b41 4141 4141
                         4141 4141 4141
09:54:40.576447 1.2.2.109.137 > OurNet.16.137: udp 50 (ttl 113, id
7363)
                         4500 004e 1cc3 0000 7111 45df 0102 026d
                         cc7e 1710 0089 0089 003a fd50 dbe0 0000
                         0001 0000 0000 0000 2043 4b41 4141 4141
                         4141 4141 4141
```

Existence: Doesn't exist. Source address is IANA reserved-9.

History: Only packets we've seen from (or to) a 1.0.0.0 space.

Techniques: Scripted netBios probe looking for open shares.

Intent: Most likely the result of another 911/Chode infection somewhere.

Targeting: Targets specifically one machine.

Severity: (4+4) – (2+5) = 1

Analysis: More 911/Chode? I'm beginning to think so. The data payload is nearly identical to the previous 137-137 scan, which both match data posted at GIAC. This one is different in that it only targets one machine, but otherwise the signature is identical to that of trace #3. I guess if all Chode wants to do is install himself, it doesn't matter what the source address is. Good thing our FW blocks incoming 137 traffic.

Trace #6

```
01:08:12.481066 210.107.239.131.53 > OurNet.12.53: 35900+ (30) (ttl 48, id 26939)
01:08:12.482683 OurNet.12.53 > 210.107.239.131.53: 35900* q: icecream.net. 1/2/2 . (140) (ttl
64, id 48915)
01:08:12.957223 210.107.233.240.1034 > OurNet.20.23: S 804971:804971(0) win 8192 <mss
1460,nop,nop,sackOK> (DF) (ttl 112, id 25856)
01:08:15.870020 210.107.233.240.1034 > OurNet.20.23: S 804971:804971(0) win 8192 <mss
1460,nop,nop,sackOK> (DF) (ttl 112, id 26112)
01:08:21.862179 210.107.233.240.1034 > OurNet.20.23: S 804971:804971(0) win 8192 <mss
1460,nop,nop,sackOK> (DF) (ttl 112, id 26368)
01:08:33.860291 210.107.233.240.1034 > OurNet.20.23: S 804971:804971(0) win 8192 <mss
1460,nop,nop,sackOK> (DF) (ttl 112, id 27136)
```

Existence: Hrmm…..another one of our Seoul friends. This time from Sejong University, how interesting.

History: None

Techniques: Looked up a host, tried to telnet to it.

Intent: Unauthorized shell access to the host.

Targeting: Targeted a single box.

Severity: (2+4) – (4+5) = -3

Analysis: Unauthorized telnet to known host attempt. Simple as they get. Someone looked up one of our hosts, attempted to telnet to it and was verily thwarted. Weird thing is, there is no traffic to any of our other hosts from this domain for at least a day before or after. If we were targeted, I would expect that they had surfed one or more of our sites first. Go figure. Oh, and the identical sequence numbers are of no consequence since the 4 packets are successively delayed by increasing amounts, most likely due to TCP stack timers on the attacking host.

Trace #7

```
First there was this:

21:11:36.191412 206.72.19.132.1114 > OurNet.3.31785: S 9135943:9135943(0) win 8192 <mss
536,nop,nop,sackOK> (DF)
21:11:36.224736 206.72.19.132.1114 > OurNet.3.31785: S 9135943:9135943(0) win 8192 <mss
536,nop,nop,sackOK> (DF)
21:11:37.762778 206.72.19.132.1114 > OurNet.3.31785: S 9135943:9135943(0) win 8192 <mss
536,nop,nop,sackOK> (DF)
21:11:41.936926 206.72.19.132.31790 > OurNet.3.31789: udp 1
21:11:41.982236 206.72.19.132.31790 > OurNet.5.31789: udp 1
21:11:42.040111 206.72.19.132.31790 > OurNet.10.31789: udp 1
21:11:42.043481 206.72.19.132.31790 > OurNet.12.31789: udp 1
21:11:42.076335 206.72.19.132.31790 > OurNet.16.31789: udp 1
21:11:42.080805 206.72.19.132.31790 > OurNet.18.31789: udp 1
21:11:42.089970 206.72.19.132.31790 > OurNet.20.31789: udp 1
21:11:42.126023 206.72.19.132.31790 > OurNet.22.31789: udp 1
21:11:42.131159 206.72.19.132.31790 > OurNet.24.31789: udp 1
21:11:49.885140 206.72.19.132.1114 > OurNet.3.31785: S 9135943:9135943(0) win 8192 <mss
536,nop,nop,sackOK> (DF)
```

Existence: The source address belongs to a local ISP of which the Net Admin happens to be a personal friend.

History: We see lots (and lots and lots) of SYNs to port 6663. Suspicious, yes, but I have other ideas.

Techniques: 2 techniques here: First is a crafted and scripted to open a TCP connection to port 31785; the other is again scripted and crafted and looking for open ports 31789 on our public net.

Intent: Looking for the Hack'a'tack Trojan.

Targeting: Specific to a box and to the network.

Severity: (3+3) – (2+5) = -1

Analysis: Script kiddie, not even original enough to look outside his small town, scanning for the Hack'a'Tack Trojan on 2 of it's common ports. At least I thought that's all it was until I also saw this (and lots of it):

```
21:00:22.046166 card1-132-lemars.pionet.net.1112 > OurNet.3.6663: S 8469085:8469085(0) win
8192  (DF)
21:00:25.031784 card1-132-lemars.pionet.net.1112 > OurNet.3.6663: S 8469085:8469085(0) win
8192  (DF)
```

```
21:00:31.028143 card1-132-lemars.pionet.net.1112 > OurNet.3.6663: S 8469085:8469085(0) win
8192  (DF)
21:00:43.026265 card1-132-lemars.pionet.net.1112 > OurNet.3.6663: S 8469085:8469085(0) win
8192  (DF)
```

I still stand by my analysis of the original trace, but this new connect attempt had me wondering. Mostly because of the same time frame, but because there were port 6663 scans from 4 other hosts in the same time. Hrmm….Spoofed addresses? More research…

21:00:22.046166 206.72.19.132.1112 > OurNet.3.6663: S 8469085:8469085(0) win 8192 <mss 536,nop,nop,sackOK> (DF)
21:00:25.031784 206.72.19.132.1112 > OurNet.3.6663: S 8469085:8469085(0) win 8192 <mss 536,nop,nop,sackOK> (DF)
21:00:31.028143 206.72.19.132.1112 > OurNet.3.6663: S 8469085:8469085(0) win 8192 <mss 536,nop,nop,sackOK> (DF)
21:00:40.523731 OurNet.3.27589 > 206.72.19.132.27744: S 553382932:553382932(0) win 8192 <mss 536,nop,nop,sackOK> (DF)
21:00:40.738286 206.72.19.132.27744 > OurNet.3.27589: S 8487766:8487766(0) ack 553382933 win 8576 <mss 536,nop,nop,sackOK> (DF)
21:00:41.036381 OurNet.3.27589 > 206.72.19.132.27744: . ack 1 win 8576 (DF)
21:00:41.040512 OurNet.3.27589 > 206.72.19.132.27744: P 1:3(2) ack 1 win 8576 (DF)

Now I think we can see what's really going on. It appears someone on our network was on ICQ or IRC (or some other similar messaging/chat agent). The 6663 scans appear to be outside hosts attempting to "chat" directly with our inside host, but the attempts are being blocked. Somehow, our inside host was notified of the chat request (not by port 6663, I assure you) and initiated the chat session herself. During the chat session is when the outside chat-member did his dirty deeds and scanned the inside host, and getting nothing from that, scanned the network. This is common in IRC, which is truly the Wild Wild West of the World Wide Web.

Trace #8

```
10:04:48.844069 pdi34.pdisrd.com.1501 > OurNet.18.25: S 6847240:6847240(0) win 8192  (DF)
10:04:52.066428 pdi34.pdisrd.com.1501 > OurNet.18.25: S 6847240:6847240(0) win 8192  (DF)
10:04:58.109807 pdi34.pdisrd.com.1501 > OurNet.18.25: S 6847240:6847240(0) win 8192  (DF)
10:05:09.818467 pdi34.pdisrd.com.1501 > OurNet.18.25: S 6847240:6847240(0) win 8192  (DF)
10:54:06.004268 pdi34.pdisrd.com.1699 > OurNet.18.25: S 9804726:9804726(0) win 8192  (DF)
10:54:08.921973 pdi34.pdisrd.com.1699 > OurNet.18.25: S 9804726:9804726(0) win 8192  (DF)
10:54:14.922907 pdi34.pdisrd.com.1699 > OurNet.18.25: S 9804726:9804726(0) win 8192  (DF)
10:54:26.926282 pdi34.pdisrd.com.1699 > OurNet.18.25: S 9804726:9804726(0) win 8192  (DF)
```

Existence: Sourced from one of our clients.

History: They've been trying to connect to sendmail on a non-sendmail host in 4 packet bursts for 4 days, but only during business hours (mostly).

Techniques: Simple SYN's to port 25, looks to be on a timer.

Intent: sendmail is infinitely exploitable. Also could be searching for mail relays.

Targeting: yes, a single host.

Severity: (2+3) – (3 + 5)

Analysis: While this could be someone searching for relays, the timing really points to a mail server trying to deliver email to the wrong address. Why it would attempt to deliver to our WWW server instead of our SMTP server when the MX records are correct is anyone's guess. A data decode of these packets, which I can't show here for obvious reasons, confirmed that indeed, someone had given out an incorrect email address (them@www.OurNet # b ######### them@www.ournetinstead of them@OurNet).

Trace #9

This trace was posted at GIAC. I didn't have a good portscan example of my own and wanted to include one.

```
Apr 1 00:06:15 cc1014244-a kernel: securityalert: tcp if=ef0 from
216.70.121.134:1119 to 24.3.21.199 on unserved port 1080
Apr 1 01:55:53 cc1014244-a kernel: securityalert: tcp if=ef0 from
210.109.56.32:3944 to 24.3.21.199 on unserved port 111
Apr 1 08:35:12 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2469
to 24.3.21.199 on unserved port 22
```

Existence: Yes and no. 1st is New York City, second is from Korea and third appears to come from the attacked host's network.

History: None.

Techniques: Host(s) is manually scanning for the known ports SOCKS proxy, SunRPC and SSH.

Intent: Appears to be scanning for available services

Targeting: Specifically targeted to a single host.

Severity: (2+4) – (4+4) = -2

Analysis: Most likely these script-kiddies have exploits for the their respective services and they're just itching to try them out. Except that last one. That looks like they were coming from the Broadcast address of the Host's subnet which would cause the host to broadcast-SynAck to the net. How this would be valuable to the attacker without his being on that net, I don't know.

Trace #10

```
12:17:02.757495 VLAN1.1.27.1025 > 255.255.255.255.41508: udp 188 (ttl 125, id 43057)
12:17:16.794222 VLAN2.21.3.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 39219)
12:17:21.955065 VLAN2.22.198.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 65187)
12:17:23.552423 VLAN2.22.161.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 51251)
12:17:24.727874 VLAN2.21.114.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 49515)
12:17:38.010877 VLAN3.20.239.1236 > 255.255.255.255.41508: udp 188 (ttl 127, id 61688)
12:17:38.756813 VLAN3.20.243.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 32607)
12:17:41.181238 VLAN3.20.126.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 25718)
12:17:41.305916 VLAN3.20.235.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 28721)
12:17:46.114786 VLAN3.20.207.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 61240)
12:17:46.360550 VLAN2.21.145.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 44581)
12:17:50.039481 VLAN3.20.190.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 26635)
12:17:54.530561 VLAN2.21.187.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 11048)
12:17:58.034240 VLAN3.20.242.1431 > 255.255.255.255.41508: udp 188 (ttl 127, id 16407)
12:17:58.959336 VLAN2.22.135.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 51008)
12:18:05.216092 VLAN3.20.185.1027 > 255.255.255.255.41508: udp 188 (ttl 127, id 22530)
12:18:10.240508 VLAN2.21.247.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 18473)
12:18:10.388242 VLAN2.22.241.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 29519)
12:18:14.826789 VLAN3.20.213.1260 > 255.255.255.255.41508: udp 188 (ttl 127, id 27352)
12:18:20.094560 VLAN3.20.196.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 48073)
12:18:21.108858 VLAN3.20.249.1026 > 255.255.255.255.41508: udp 188 (ttl 127, id 15665)
12:18:22.019553 VLAN2.20.190.1026 > VLAN2.255.255.41508: udp 188 (ttl 128, id 19439)
12:18:25.604442 VLAN2.22.137.1386 > VLAN2.255.255.41508: udp 188 (ttl 128, id 40056)
12:18:26.801405 VLAN2.20.177.1381 > VLAN2.255.255.41508: udp 188 (ttl 128, id 31137)
```

Existence: All address are part of our internal networks

History: I've seen traffic like this on our network before, but prior to the IDIC I didn't give it much thought.

Techniques: The multiple source addresses would indicate spoofed source addresses.

Intent: Looks like they are attempting to DoS our internal networks.

Targeting: Targets the entire domain.

Severity: (4+5) – (4+2) = 3

Analysis: My initial thought was that someone was getting on our network and spoofing many internal addresses. Through those spoofed addresses they were sending UDP packets to the broadcast domain hoping the resulting flood of return packets would kill the spoofed box. Alternatively, they could have spoofed the broadcast addresses and what shows here are the responses. Either way, more data is needed.

And I found it. Port 41508 is used by Cheyenne Backup software (now part of CA's Unicenter TNG) to keep track of backup servers and subnets on the network. Apparently our [insert adjective here] system admins loaded the backup server on every NT box on our network and they wile away the time chatting to each other AND THE ENTIRE BROADCAST DOMAIN. A talk is scheduled.

The real scary part here is some, but not all, of the "servers" learned about the network(s) between our firewall and gateway and were broadcasting that as well. Nice bits if information for anyone who cares to pick them up.

Trace #11

```
15:07:01.930209 207.199.77.109.137 > OurNet.2.137: udp 50
15:07:03.432631 207.199.77.109.137 > OurNet.2.137: udp 50
15:07:04.927549 207.199.77.109.137 > OurNet.2.137: udp 50
15:07:13.956333 207.199.77.109.137 > OurNet.3.137: udp 50
15:07:15.457201 207.199.77.109.137 > OurNet.3.137: udp 50
15:07:16.957437 207.199.77.109.137 > OurNet.3.137: udp 50
15:07:32.016140 207.199.77.109.137 > OurNet.5.137: udp 50
15:07:33.516954 207.199.77.109.137 > OurNet.5.137: udp 50
15:07:35.016739 207.199.77.109.137 > OurNet.5.137: udp 50
15:08:26.151952 207.199.77.109.137 > OurNet.10.137: udp 50
15:08:27.651909 207.199.77.109.137 > OurNet.10.137: udp 50
15:08:29.152098 207.199.77.109.137 > OurNet.10.137: udp 50
15:08:47.208735 207.199.77.109.137 > OurNet.12.137: udp 50
15:08:48.708584 207.199.77.109.137 > OurNet.12.137: udp 50
15:08:50.211181 207.199.77.109.137 > OurNet.12.137: udp 50
15:09:23.318535 207.199.77.109.137 > OurNet.16.137: udp 50
15:09:24.816508 207.199.77.109.137 > OurNet.16.137: udp 50
15:09:26.316877 207.199.77.109.137 > OurNet.16.137: udp 50
15:09:41.381164 207.199.77.109.137 > OurNet.18.137: udp 50
15:09:42.876634 207.199.77.109.137 > OurNet.18.137: udp 50
15:09:44.376434 207.199.77.109.137 > OurNet.18.137: udp 50
15:09:59.440780 207.199.77.109.137 > OurNet.20.137: udp 50
15:10:00.938565 207.199.77.109.137 > OurNet.20.137: udp 50
15:10:02.438215 207.199.77.109.137 > OurNet.20.137: udp 50
15:10:17.497543 207.199.77.109.137 > OurNet.22.137: udp 50
15:10:18.997166 207.199.77.109.137 > OurNet.22.137: udp 50
15:10:20.498429 207.199.77.109.137 > OurNet.22.137: udp 50
15:10:40.063472 207.199.77.109.137 > OurNet.24.137: udp 50
15:10:41.556825 207.199.77.109.137 > OurNet.24.137: udp 50
```

```
15:10:43.057931 207.199.77.109.137 > OurNet.24.137: udp 50
15:10:58.110592 207.199.77.109.137 > OurNet.26.137: udp 50
15:10:59.610129 207.199.77.109.137 > OurNet.26.137: udp 50
15:11:01.114292 207.199.77.109.137 > OurNet.26.137: udp 50
```

I'm throwing this one in for good measure. Looks like another Chode/911 infection scan. How do they scan only our trafficked hosts, though? Do I have SHADOW tuned incorrectly? I need to put a strong analyzer on this.