*** Northcutt - solid use of an analysis process, research on the source addresses, a strong sense of history. You don't see that many land attacks in the wild, good catch! 78 *

GIAC Certified Intrusion Analysts (GCIA) Practical Exam Ten Detects with Analysis

SANS 2000 Orlando, Florida
Intrusion Detection Immersion Curriculum
March 21 – 25, 2000

Terry L. Henderson

April 19, 2000

As a member of a very small IDS team working for a large corporation, I am limited to the amount of data I can get access to or give without breaking the corporate rules. The IDS we use is pretty limited in the amount of detail that is captured and I am not allowed to get firewall or router logs. We must coordinate any investigation of anomalies with the groups who manage those tools in order to ensure penetration failure.

We are a worldwide organization with multiple mainframes and hundreds of hosts with almost every O/S on the market. We have multiple firewalls and remote access servers deployed around the world. The firewalls and routers are centrally managed with strict policies and access rules and the remote access servers must follow central policy rules as well.

The IDS sniffers are located strategically around the firewalls and RAS, as well as clustered at all the choke points in the central network operations center. We have other intrusion tools deployed all over our network including several host-based products.

We have had very few detects since deployment and have never had anything out of the 'ordinary'. But, with the experience I have had and the excellent education from the Sans Organization, other organizations, and my love of this field, I feel I am fully qualified to detect and analyze any anomaly. We have either been very lucky or we have our defenses tied down so tight we squeak. My guess is luck. Either way, not much has made it through as you can see.
Most of the detects were taken from our DMZ and all have been very sanitized.
a.b.c.1 = attacker x.y.z.2 = my network

DETECT 1 – Port Scanner

```
18:19:12.426937 a.b.c.1.3075 > x.y.z.2.21: S 338936:5338936(0) win 8192
(DF)
18:19:12.433432 a.b.c.1.3076 > x.y.z.2.23: S 5338948:5338948(0) win 8192
(DF)
18:19:12.444009 a.b.c.1.3077 > x.y.z.2.25: S 5338996:5338996(0) win 8192
(DF)
18:19:12.453699 a.b.c.1.3078 > x.y.z.2.53: S 5338984:5338984(0) win 8192
(DF)
18:19:12.464582 a.b.c.1.3079 > x.y.z.2.111: S 338976:5338976(0) win 8192
(DF)
18:19:12.473757 a.b.c.1.3080 > x.y.z.2.137: S 5339076:5339076(0) win
8192  (DF)
18:19:12.484269 a.b.c.1.3081 > x.y.z.2.11: S 5339020:5339020(0) win 8192
(DF)
18:19:12.493366 a.b.c.1.3082 > x.y.z.2.19: S 5339160:5339160(0) win 8192
(DF)
```

This capture shows the attacker performing a simple TCP Port Scan using SYN packets. The source port is clearly crafted while the destination ports are some of the old standard well known ones. Obviously the attacker would receive a SYN/ACK response from any open ports and RESET/ACK's from the closed ones giving the knowledge of which ports would allow possible access to this host.

Active Targeting: Yes

History: This is the only time I have seen any reconnaissance attempt from this source.

Technique: This was a quick scan for a few well-known ports. (ftp, telnet, smtp, dns, rpc/portmapper, netbios-name service, systat, and chargen) I don't know if the intention was to return later for other ports, but I never saw any indication of this (even days later).

Intent: The attacker is checking for vulnerabilities and would have perhaps taken a shot at some of them had he received a positive response from our host.

Evaluation: None of these ports are open on this host so the result of the reconnaissance was fruitless. Performed reverse lookup and trace route on source address and it will be monitored for future activity.

DETECT 2 – Script Kiddie - Log from commercial IDS

```
www Perl interpreter attack   a.b.c.1.0   x.y.z.2.80  09:55:09 count = 1
www PHF         a.b.c.1.0   x.y.z.2.80   09:55:25   count = 1
www Test-CGI Bug   a.b.c.1.0   x.y.z.2.80   09:55:34   count = 5
www Php View File Bug   a.b.c.1.0   x.y.z.2.80   09:55:40   count = 2
www Handler CGI Bug   a.b.c.1.0   x.y.z.2.80   09:55:45   count = 1
www CGI Wrap Bug   a.b.c.1.0   x.y.z.2.80   09:55:45   count = 2
www Glimpse Server Attack   a.b.c.1.0   x.y.z.2.80   09:55:45   count =
1
www WebSendmail Bug   a.b.c.1.0   x.y.z.2.80   09:55:55   count = 1
www NPH-Test-CGI Bug   a.b.c.1.0   x.y.z.2.80   09:55:55   count = 2
Website Win-c-sample Buffer Overflow   a.b.c.1.0   x.y.z.2.80   09:56:08
count = 2
www Webdist Bug   a.b.c.1.0   x.y.z.2.80   09:56:15   count = 1
www IIS New DSN Attack   a.b.c.1.0   x.y.z.2.80   09:56:33   count = 1
www CGI-bin   a.b.c.1.0   x.y.z.2.80   09:56:56   count = 20
```

I thought it would be interesting to show what a commercial IDS shows when it matches signatures. This capture shows the attacker apparently performing a script to try several attacks on our web server. I find it interesting that the script tries 20 of the CGI-bin attacks, obviously going after several of the common files (etc/password, etc..). The script ran for almost two minutes. The challenge when watching icons from a commercial IDS is recognizing these patterns that are mixed in with all the false positives.

Active Targeting: Yes

History: This type of script is fairly common and has occurred several times from different ISP’s around the world.

Technique: This was a fairly slow hit and run script technique probably found on the web and maybe executed just for giggles. I feel that the attacker was probably not knowledgeable yet of crafting his own attacks or perhaps he is just lazy.

Intent: This script tried some pretty good attacks and obviously the sender didn’t try to keep his attack a secret. Otherwise, I believe each attempt would have been spread out to make it more difficult for me to notice the pattern. Was he smart enough to take advantage of information that was supposed to be returned to him? Don’t know, but there was obvious intent. A script kiddie is looking for the easy kill. They focus on a small number of exploits, and then searching the entire Internet for that exploit.

Evaluation: None of these attacks succeeded and the web server was not compromised. The number of tries and the almost two minute time frame made the attack fairly easy to detect which makes me believe this is a novice. The ISP was contacted, since this attacker was not that too smart, and we thought we might be able to gather data on him. The ISP was unwilling or unable to help us but we have since kept watch on that source address.

DETECT 3 – Smurf Attack

```
09:15:22 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:22 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:22 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:23 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:23 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:23 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:25 a.b.c.1 > x.y.z.255: icmp: echo request
09:15:25 a.b.c.1 > x.y.z.255: icmp: echo request
```

This capture came from the dirty side of our firewall and is the standard Smurf that still shows up quite frequently. It still seems to be a very popular attack although I can't imagine anyone being vulnerable to this anymore. Obviously the attacker thinks networks are still susceptible so maybe I'm all wet.

Active Targeting: Yes

History: Happens almost daily from different, apparently spoofed, sources.

Technique: Simple pings to a DMZ server broadcast address. Nothing fancy here but potential slow down of network could cause serious problems.

Intent: The perpetrator was apparently trying for a slowdown on our network as well as the spoofed source.

Evaluation: It just so happened that the source address used on this attack was one of our major competitors. I'm sure they would have enjoyed our echo replies but since we don't allow pings through this firewall, we never sent any replies. The perp was evidently having fun by trying to start a feud between the companies. No other attacks appeared during this detect or afterwards as you would possibly expect if he was trying something different like a hijack. Also, broadcasts are not allowed on this server operating system so he would not have had an effect on us even if the firewall had let him in.

DETECT 4 – Network Mapping

```
00:20:47.215225 a.b.c.1 > x.y.z.2: icmp: echo request
00:20:47.218810 a.b.c.1 > x.y.z.3: icmp: echo request
00:20:47.222216 a.b.c.1 > x.y.z.4: icmp: echo request
00:20:47.224732 a.b.c.1 > x.y.z.5: icmp: echo request
00:20:47.227509 a.b.c.1 > x.y.z.6: icmp: echo request
00:20:47.231859 a.b.c.1 > x.y.z.7: icmp: echo request
00:20:47.233331 a.b.c.1 > x.y.z.8: icmp: echo request
00:20:47.238265 a.b.c.1 > x.y.z.9: icmp: echo request
```

This capture came from the dirty side of the firewall and is the classic fast single host network mapper. He sent about twenty pings every fifteen minutes for about forty-five minutes before giving up. The source address is real but could certainly not be the real source. The attacker would have received an echo reply from each host listening.

Active Targeting: Yes

History: This type of reconnaissance is sometimes seen after an attempt to ping using broadcast addresses has failed.

Technique: This was a quick bursts of pings and then waiting for fifteen minutes to try and keep from being noticed. It had to be sent as a script to move at burst speed like this and probably has a timer also to perform the start stop task and keep up with the destination addresses. There were no other anomalies from this source. Don't know if he gave up or changed sources.

Intent: The attacker is trying to check for active host on this network and I'm sure he would have taken a second step had he received the expected results. He wasn't pinging us for the fun of it.

Evaluation: This firewall blocks pings so there is no reason to worry about this type of reconnaissance attempt. This would explain why he gave up after forty-five minutes. The source was an ISP in Brazil so we did not bother calling. We checked the clean side of the firewall and made sure nothing got through. The automated burst and delay tactic seems like a good way to ping all the way through a class c from an unattended PC. It does offer a small measure of invisibility while still getting the job done.

DETECT 5 – Trojan Scan

```
20:32:01.122735 a.b.c.1.34620 > x.y.z.2.31337: S 4784849:4784849(0) win
8192  (DF)
20:32:01.134637 a.b.c.1.34621 > x.y.z.2.137: S 4784900:4784900(0) win
8192  (DF)
20:32:01.146503 a.b.c.1.34622 > x.y.z.2.2140: S 4822207:4822207(0) win
8192  (DF)
20:32:01.158497 a.b.c.1.34623 > x.y.z.2.139: S 4948487:4948487(0) win
8192  (DF)
20:32:01.160383 a.b.c.1.34624 > x.y.z.2.31335: S 4962222:4962222(0) win
8192  (DF)
20:32:01.172255 a.b.c.1.34625 > x.y.z.2.3150: S 5255444:5255444(0) win
8192  (DF)
```

This capture from the DMZ shows the attacker performing a syn port scan for common trojan ports on one of our DMZ servers hoping to find something he can take advantage of.

Active Targeting: Yes

History: This is not as common as pings for trojans on the dirty side of the firewall that never make it through, but does happen occasionally. Records are kept on the offending source address.

Technique: This was a quick scan for a few well-known trojan ports. (back orifice, trin00, netbios, deep throat). Nothing fancy and certainly not a complete list, just boldly looking for installed programs on one host.

Intent: The attacker is hoping to find trojans that he can take advantage of in order to gain access or cause havoc to my network. Obvious intent here.

Evaluation: The source ports were apparently crafted but the syn's were bold enough that the attacker wasn't trying to hide. Had the perpetrator been able to find one of these ports active on this host, he is basically in. Since he was targeting a DMZ host, the network would still not be available to him, but he may have been able to wreak havoc in the DMZ and possibly found other holes. The source was an ISP in California and a quick call told us we would not be helped (we'll check it out and get back with you). We regularly scan servers for viruses and trojans and keep our antivirus product updated weekly to ensure we are not running any of these programs on our equipment.

DETECT 6 – Syn Flood

```
17:47:16.727461 a.b.c.1.907 > x.y.z.2.139: S 3037242:3037242(0) win 512
(DF)
17:47:16.795353 a.b.c.1.908 > x.y.z.2.139: S 4984920:4984920(0) win 512
(DF)
17:47:17.083214 a.b.c.1.909 > x.y.z.2.139: S 5500161:5500161(0) win 512
(DF)
17:47:17.120986 a.b.c.1.910 > x.y.z.2.139: S 5948387:5948387(0) win 512
(DF)
17:47:17.192215 a.b.c.1.911 > x.y.z.2.139: S 6207914:6207914(0) win 512
(DF)
17:47:17.234978 a.b.c.1.912 > x.y.z.2.139: S 6854777:6854777(0) win 512
(DF)
17:47:17.325576 a.b.c.1.913 > x.y.z.2.139: S 7298984:7298984(0) win 512
(DF)
17:47:17.461984 a.b.c.1.914 > x.y.z.2.139: S 7522255:7522255(0) win 512
(DF)
17:47:17.729326 a.b.c.1.915 > x.y.z.2.139: S 7965844:7965844(0) win 512
(DF)
17:47:18.097852 a.b.c.1.916 > x.y.z.2.139: S 8207098:8207098(0) win 512
(DF)
```

This capture shows the attacker performing a syn flood directed towards one of our servers in the DMZ.

Active Targeting: Yes

History: This occurs frequently but most of the time they appear to be false positives. This one stood out as a little different from the others and as far as I know, it has never happened before or since.

Technique: This was a little strange to me since it was tcp going after port 139 on a UNIX server??? Was the attacker confused or just me? The source ip and port are crafted and he was obviously trying to tie up this server's connection queue.

Intent: This was certainly intentional since everything was crafted. I can't comment on the port he was going after. Maybe I just don't understand the connection with UNIX and port 139. Anyway, it looks like the intent was there. I don't have any clues as to what the perp's next move was going to be, but I saw nothing further that could be associated with this anomaly.

Evaluation: Crafted syn packets sent to a server with non-existent source ip's can only be sent with the intent of denial of service on that host. Had this attack been successful, the host would have been tied up for a period of time or locked up permanently if unable to handle the waiting period. The source was untraceable and nothing appeared to be affected on the server.

DETECT 7 – Land Attack

```
21:49:02.016552 x.y.z.2.23 > x.y.z.2.23: S 8099572:8099572(0) win 4096
(DF)
21:49:02.235109 x.y.z.2.23 > x.y.z.2.23: S 4011917:4011917(0) win 4096
(DF)
21:49:03.083599 x.y.z.2.23 > x.y.z.2.23: S 4430816:4430816(0) win 4096
(DF)
21:49:03.136842 x.y.z.2.23 > x.y.z.2.23: S 5300196:5300196(0) win 4096
(DF)
21:49:04.105444 x.y.z.2.23 > x.y.z.2.23: S 0355992:0355992(0) win 4096
(DF)
```

This capture from the dirty side of the firewall shows the attacker performing a land attack directed towards a DMZ server.

Active Targeting: Yes

History: This is the only occurrence I have seen of this attack. Perhaps it is becoming passé but I thought it was somewhat interesting.

Technique: This is your standard old land attack. Evidently the attacker knew enough about our network to get the address of a DMZ server to try to have his way with us. This may have come from the good old targa program since it sent a group of five packets.

Intent: This was certainly intentional since the packets were crafted. Obviously the intent was to lock up a host in our DMZ and cause denial of service.

Evaluation: Crafted syn packets sent to a server with the same source ip as the destination ip is only sent with the intent of denial of service on that host. Had this attack been successful, the host could have been locked up permanently and would probably have needed to be rebooted. Our routers are configured to extract these anomalies from our network so they cannot reach the intended target. The host was checked to make sure it was unaffected even though the packets were dropped before they reached their destination. Maybe some paranoia setting in.

DETECT 8 – SunRPC/Portmapper

```
12:02:16.443927 a.b.c.1.15034 > x.y.z.2.111: S 2295505:2295505(0) win
1024  (DF)
12:02:16.469802 a.b.c.1.15065 > x.y.z.6.111: S 2400694:2400694(0) win
1024  (DF)
12:02:16.503336 a.b.c.1.15096 > x.y.z.4.111: S 3877762:3877762(0) win
1024  (DF)
12:02:16.526571 a.b.c.1.15127 > x.y.z.3.111: S 4100338:4100338(0) win
1024  (DF)
12:02:16.558806 a.b.c.1.15158 > x.y.z.5.111: S 4473892:4473892(0) win
1024  (DF)
12:02:16.600339 a.b.c.1.15189 > x.y.z.7.111: S 4826784:4826784(0) win
1024  (DF)
```

This capture came from inside our network and shows the attacker performing a scan looking for the SunRPC/Portmapper port.

Active Targeting: Yes

History: This has become a fairly common probe on our network but never from the same source. Since we have all types of operating systems, the perp can be probing for the portmapper or remote procedure call feature.

Technique: This perpetrator came inside scanning several ip's for the Sun remote procedure call or portmapper port. The source port increments by 31 for each packet and was crafted. These attacks generally come in fast bursts of five to ten packets.

Intent: This was certainly intentional since the packets were crafted. The obvious intent was to find the remote procedure call to execute programs of their choosing or ask the portmapper to show the active ports on the host.

Evaluation: Crafted syn packets sent to several servers using the same source ip with incrementing port numbers. The destination ip's are being checked for active port 111 to execute programs or map the host ports. None of the hosts contacted responded positively so the perpetrator left empty handed. The source ip was an ISP in Finland so we didn't bother calling for their assistance.

DETECT 9 – DNS Scan

```
03:10:52.243587 a.b.c.1.3050 > x.y.z.2.53: S 3562477:3562477(0) win 512
(DF)
03:12:55.425633 a.b.c.1.3051 > x.y.z.3.53: S 0278447:0278447(0) win 512
(DF)
03:14:59.109576 a.b.c.1.3052 > x.y.z.4.53: S 6878924:6878924(0) win 512
(DF)
03:17:04.882634 a.b.c.1.3053 > x.y.z.5.53: S 2556473:2556473(0) win 512
(DF)
03:19:07.863710 a.b.c.1.3054 > x.y.z.6.53: S 2994782:2994782(0) win 512
(DF)
03:21:10.758592 a.b.c.1.3055 > x.y.z.7.53: S 4956432:4956432(0) win 512
(DF)
03:23:14.343467 a.b.c.1.3056 > x.y.z.8.53: S 5493305:5493305(0) win 512
(DF)
03:25:17.638588 a.b.c.1.3057 > x.y.z.9.53: S 4628334:4628334(0) win 512
(DF)
```

This capture came from inside our network and shows the attacker performing a scan looking for a DNS port on any host.

Active Targeting: Yes

History: This is a fairly common probe on our network and varies slightly in technique. We do not see the same source ip except in extended scans that go on for several hours.

Technique: This perpetrator got inside in the wee hours of the morning and scanned several different ip addresses looking for the DNS port active. This particular attacker used a pattern of sending about ten packets, waiting ten minutes and sending about ten more. This was apparently an unattended program or script he was running to automatically scan through one of our class c's. The source ports were crafted and the packets were sent slowly to avoid detection.

Intent: The intent to map our network and gather DNS information is certainly there.

Evaluation: Crafted syn packets sent to progressive server ip's using the same source ip with incrementing port numbers. The destination ip's are being checked for active port 53 to extract DNS information or try other exploits. At the same time, he is mapping the class c network. This particular scan stopped after a couple of hours which indicated to me that he was not receiving what he was expecting or maybe was afraid of being noticed. If he had received a DNS hit, then he may have tried downloading the zone file gathering lots of damaging information. The source ip was another ISP in Finland so we didn't bother calling for their assistance.

DETECT 10 – QueSO

```
06:20:06.243587 a.b.c.1.8270 > x.y.z.2.80: S 2357489:2357489(0) ack 0
win 4660
06:20:06.244070 a.b.c.1.8271 > x.y.z.2.80: F 0278447:0278447(0) win 4660
06:20:06.250288 a.b.c.1.8272 > x.y.z.2.80: F 6878924:6878924(0) ack 0
win 4660
06:20:06.250370 a.b.c.1.8273 > x.y.z.2.80: SF 2556473:2556473(0) win
4660
06:20:06.259788 a.b.c.1.8274 > x.y.z.2.80: P win 4660
06:20:06.268506 a.b.c.1.8275 > x.y.z.2.80: S 4956432:4956432(0) win 4660
```

This capture in the DMZ looks strongly like someone running QueSO against one of our webservers.

Active Targeting: Yes

History: This was not seen very often but we used to get it occasionally. It was never the same source address.

Technique: This looks like classic QueSO to me. I did not see the original handshake on the log that tested to see if the port was listening but it could have been done days before. QueSO crafts these packets when you point it to an address then in a flash it describes the operating system of the target machine.

Intent: The intent seems clearly to get the webserver to answer the above packets and then determine the operating system and perhaps the version. Exploit of the known holes in the O/S seemed imminent.

Evaluation: I did not show the webserver responses because if you are familiar with QueSO, you would know my webserver. Crafted malformed packets, in this order, sent to a host will give QueSO the information it needs to determine the O/S by the responses it gets back. The perpetrator got exactly what he was looking for. We asked for a new firewall rule to statefully handle this and have not seen it again. Since it was an old detect, the ip address was not contacted but we made double sure the server was properly hardened anyway.