



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Intrusion Detection In-Depth (Security 503)"
at <http://www.giac.org/registration/gcia>

***** **SANS GCIA PRACTICAL** *****

Assignment 1 : Network Detects

DETECT # :

Detect 1

```
May 20 06:58:06 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.104.255 (8/0), 1 packet
May 20 06:58:11 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.105.255 (8/0), 1 packet
May 20 06:58:31 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.109.255 (8/0), 1 packet
May 20 06:58:36 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.110.255 (8/0), 1 packet
May 20 06:58:42 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.111.255 (8/0), 1 packet
May 20 06:58:47 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.112.255 (8/0), 1 packet
May 20 06:58:52 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.114.255 (8/0), 1 packet
May 20 06:58:57 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.115.255 (8/0), 1 packet
May 20 06:59:02 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.115.255 (8/0), 1 packet
May 20 06:59:07 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.116.255 (8/0), 1 packet
May 20 06:59:12 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.117.255 (8/0), 1 packet
May 20 06:59:18 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.118.255 (8/0), 1 packet
May 20 06:59:23 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.119.255 (8/0), 1 packet
May 20 06:59:28 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.120.255 (8/0), 1 packet
May 20 06:59:33 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.121.255 (8/0), 1 packet
May 20 06:59:38 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.122.255 (8/0), 1 packet
May 20 06:59:43 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.123.255 (8/0), 1 packet
May 20 06:59:48 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.124.255 (8/0), 1 packet
.....
May 20 07:11:04 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.y.255.255 (8/0), 1 packet
May 20 07:11:06 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.z.0.255 (8/0), 1 packet
May 20 07:11:12 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.z.1.255 (8/0), 1 packet
```

```
May 20 07:11:22 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.z.4.0 (8/0), 1 packet  
May 20 07:11:23 %SEC-6-IPACCESSLOGDP list 101 denied icmp 24.1.244.50 -> x.z.4.255 (8/0), 1 packet
```

SOURCE OF TRACE :

Client Network

DETECT WAS GENERATED BY :

This detect was generated by a Cisco router that is configured as a security or filtering router, sitting between the Internet and the client's DMZ. List 101 is a Cisco Extended Access Control List (ACL), specifically filtering inbound traffic from the Internet. This part of the ACL rule-set is designed to block and log inbound traffic to net address x.x.x.0 or broadcast address x.x.x.255.

PROBABILITY THE SOURCE ADDRESS WAS SPOOFED:

The probability that this source is not spoofed is very high. The reason for this belief is because 24.x.y.z is an IP from the @Home network and there are lots of wanna-be crackers using this cable modem service to initiate all types of attacks. If the source were spoofed, it would probably be another cable modem user usurping a fellow cable modem user on a common collision domain. And as such, with the current information from the security router alone, there is not enough information to determine spoofing. Correlation can aid the analyst, who knows his network, in quickly determining such information.

DESCRIPTION OF ATTACK :

This attack is against the broadcast and legacy addresses of supposedly existing subnets. The tool is some sort of scanner, which probably looks for responses from hosts that reply back. The tool may also have a mechanism to avoid a self imposed DoS based on too many replies coming back too soon.... However due to the fact that this is a cable modem

User and depending on the time that the attacker chooses to come online, he could have considerable bandwidth. Although not explicitly shown and to protect the guilty, the attacker is attempting to completely map a Class-A IP address Space. If the attacker were attempting a DoS attack against a specific host then it would be a Smurf style attack. Please see the CERT Advisory that explicitly defines this sort of attack:

<http://www.cert.org/advisories/CA-98.01.smurf.html>

ATTACK MECHANISM :

By sending icmp echo requests [the (8/0) states that the icmp packet was a ping. Please see TCP/IP Illustrated Volume 1, Chapter 6, Section 6.2 and Chapter 7] to the broadcast or net addresses of subnets, one is able to determine what hosts exist on that particular subnet. This is a first level attempt at reconnaissance. As the traces show, an attacker is sequentially stepping through each subnet, yet there are some subnets that have not been pinged. Could this be that the attacker is already aware of these subnets? And the pinging of network 115 twice seems odd as well. Also, as can be seen in the last five traces, the attacker has attempted to map the whole 'y' subnetworks and then moves to map the 'z' subnetworks. This activity persisted for 6 hours.

CORRELATIONS :

This type of attack is classic in its nature, and shows that the attacker has some understanding of networking. **CERT Advisory CA-98.01.smurf.html** defines this attack, as well as its reconnaissance capabilities. Also, please check out the Excellent FAQ by Robert Graham which talks about what one sees at the firewall: <http://www.robertgraham.com>

EVIDENCE OF ACTIVE TARGETING :

Active targeting of specific hosts is not the case here, but active targeting against specific subnets is very evident.

SEVERITY :

Criticality: Core Router > 5
Lethality: Confidentiality attack > 2
System Countermeasures: Modern Cisco IOS > 5
Network Countermeasures: Validated Restrictive Firewall > 5

$(\text{Criticality} + \text{Lethality}) - (\text{System} + \text{Network Countermeasures}) = \text{Severity}$ $(5+2)-(5+5) = -3$ Very Low Risk...But
“If a vulnerability is never targeted, is it really a vulnerability”?

DEFENSE RECOMMENDATION :

By using packet-filtering at the network level, an enterprise can reduce the capability of attackers mapping their networks, scanning their hosts for vulnerabilities, and commencing exploits on those exposed hosts. Cisco routers achieve packet-filtering capabilities by implementing ACLs. As mentioned previously, ACL 101 blocks and logs specific incoming types of traffic that has been defined as harmful. Thus the recommendation is to have incoming as well as outgoing ACLs that block traffic deemed anomalous by the security analyst.

MULTIPLE CHOICE TEST QUESTION :

Based on the network traces shown above, what is this possible evidence of?

- (a) pinging broadcast addresses
- (b) Smurf attack
- (c) reconnaissance
- (d) all of the above

The answer is d... more information would be needed... i.e. **correlation** to make the exact call... thus the analyst needs to consider all cases and cancel out the possibilities through correlation.

DETECT # :

Detect 2

```
May 22 12:09:08 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55819) -> x.y.127.192 (3128), 1 packet
May 22 12:09:09 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55820) -> x.y.127.192 (3128), 1 packet
May 22 12:10:27 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (6667), 1 packet
May 22 12:11:12 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (3128), 1 packet
May 22 12:11:17 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (27665), 1 packet
May 22 12:11:27 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55822) -> x.y.127.192 (4), 1 packet
May 22 12:11:28 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (4), 1 packet
May 22 12:11:34 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55822) -> x.y.127.192 (6969), 1 packet
May 22 12:11:35 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (6969), 1 packet
May 22 12:12:03 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55822) -> x.y.127.192 (6666), 1 packet
May 22 12:12:42 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55822) -> x.y.127.192 (5), 1 packet
May 22 12:12:43 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (5), 1 packet
May 22 12:12:47 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55823) -> x.y.127.192 (31337), 1 packet
May 22 12:13:26 %SEC-6-IPACCESSLOGDP list 101 denied tcp 24.28.140.221(55822) -> x.y.127.192 (79), 1 packet
```

SOURCE OF TRACE :

Client Network

DETECT WAS GENERATED BY :

This detect was generated by a Cisco router that is configured as a security or filtering router, sitting between the Internet and the client's DMZ. List 101 is a Cisco Extended Access Control List (ACL), specifically filtering inbound traffic from the Internet. This part of the ACL rule-set is designed to block and log inbound traffic to known Trojan ports.

PROBABILITY THE SOURCE ADDRESS WAS SPOOFED:

The probability that this source is not spoofed is very high. The reason for this belief is because 24.x.y.z is an IP from the @Home network and there are lots of wanna-be crackers using this cable modem service to initiate all types of attacks. If the source were spoofed, it would probably be another cable modem user usurping a fellow cable modem user on a common collision domain. And as such, with the current information from the security router alone, there is not enough information to determine spoofing. Correlation can aid the analyst, who knows his network, in quickly determining such information.

DESCRIPTION OF ATTACK :

This attack is a probe for Trojans, specifically attempting to invoke them on a host suspected of having these popular backdoors. Let's observe the different backdoors that the attacker scans on host x.y.127.192:

- Two RingZero probes > port 3128/tcp
- A Trojan ScheduleAgent probe or an IRC probe with a potential backdoor > port 6667tcp
- A RingZero probe > port 3128/tcp
- A Trinoo Agent probe > port 27665/tcp
- Two probes at port 4; currently there are no well-known services that the author knows of which would operate on this port; hence traffic to this port is considered anomalous. It is also known that the probe tool sscan checks ports 1-5/tcp.
- Two probes for either Gatecrasher, Priority, IRC3, or NetController > port 6969
- A probe for IRC or Napster; there could be new Trojans which use Napster or IRC > port 6666/tcp
- Two probes at port 5; currently there are no well-known services that the author knows of which would operate on this port; hence traffic to this port is considered anomalous. It is also known that the probe tool sscan checks ports 1-5/tcp.

- A Back Orifice probe > port 31337/tcp
- A probe for finger which can provide an attacker with intimate knowledge of users on a host > port 79/tcp

ATTACK MECHANISM and CORRELATIONS :

This attack strongly represents an in-depth reconnaissance effort using the tool sscan. We previously saw that we had probes to ports 4 and 5; this is evidence of an sscan probe and the tool has been configured to probe Trojan ports:

The sscan tool performs probes against victim hosts to identify services, which may potentially be vulnerable to exploitation. Though sscan itself does not attempt to exploit vulnerabilities, it can be configured to automatically execute scripts of commands that can be maliciously crafted to exploit vulnerabilities. Thus, it is possible for an unpredictable set of attacks to be mounted against a victim site in conjunction with the sscan probes.

<http://www.codetalker.com/advisories/cert/in-99-01.html>

Additionally we may possibly conclude that sscan uses ports 55819-55823 to scan potential victims for Trojans. More research is required to validate this hypothesis.

EVIDENCE OF ACTIVE TARGETING :

This is evidence of active targeting against specific host: x.y.127.192

SEVERITY :

Criticality: Core Router > 5

Lethality: Attacker can gain root access across the net > 5

System Countermeasures: Modern Cisco IOS > 5

Network Countermeasures: Validated Restrictive Firewall > 5

(Criticality + Lethality) – (System + Network Countermeasures) = Severity (5+5)-(5+5) = 0 Very Low Risk...But
“If a vulnerability is never targeted, is it really a vulnerability”?

DEFENSE RECOMMENDATION :

By using packet-filtering at the network level, an enterprise can reduce the capability of attackers mapping their networks, scanning their hosts for vulnerabilities, and commencing exploits on those exposed hosts. Cisco routers achieve packet-filtering capabilities by implementing ACLs. As mentioned previously, ACL 101 blocks and logs specific incoming types of traffic that has been defined as harmful. Thus the recommendation is to have incoming as well as outgoing ACLs that block traffic deemed anomalous by the security analyst.

MULTIPLE CHOICE TEST QUESTION :

Probes to ports 1-5 indicate what?

1. Regular network traffic
2. Searching for Napster Servers
3. Scanning hosts using sscan
4. None of the above

The answer is 3, based on the 2 sets of probes to ports 4 and 5; to really understand why this seems to be the conclusion please check: <http://www.codetalker.com/advisories/cert/in-99-01.html>

DETECT # :

Detect 3
July 25 02:30:27.498530 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80->x.y.35.101,2074 PR tcp len 20 44 -AS
July 25 05:51:05.035830 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.46,2074 PR tcp len 20 44 -AS
July 25 06:59:06.507303 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.120,2744 PR tcp len 20 44 -AS
July 25 08:31:20.790704 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.79,1611 PR tcp len 20 44 -AS
July 25 09:52:53.243289 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.88,1638 PR tcp len 20 44 -AS
July 25 10:10:15.109860 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.111,1482 PR tcp len 20 44 -AS
July 25 12:48:11.466086 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.57,1285 PR tcp len 20 44 -AS
July 25 13:45:09.876969 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.65,2152 PR tcp len 20 44 -AS
July 25 14:54:09.984609 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.69,2437 PR tcp len 20 44 -AS
July 25 14:54:12.393603 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.69,2437 PR tcp len 20 44 -AS
July 25 17:13:56.878704 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.40,2345 PR tcp len 20 44 -AS
July 25 20:02:50.987704 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.51,1656 PR tcp len 20 44 -AS
July 25 20:45:37.343595 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.43,1152 PR tcp len 20 44 -AS
July 25 22:31:34.400053 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.95,1354 PR tcp len 20 44 -AS
July 26 00:14:11.123147 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.92,2306 PR tcp len 20 44 -AS
July 26 01:23:59.232340 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.91,2267 PR tcp len 20 44 -AS
July 26 01:35:19.848921 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.112,1605 PR tcp len 20 44 -AS
July 26 02:41:15.869036 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.95,2276 PR tcp len 20 44 -AS
July 26 02:48:31.523615 dragon ipmon[502] fxp0 @0:23 b 207.153.241.222,80 -> x.y.35.42,1236 PR tcp len 20 44 -AS

SOURCE OF TRACE :

Protected Lab Network

DETECT WAS GENERATED BY :

IP Filter firewall package protecting a lab network. The OS is Open BSD 2.6 on an Intel Machine. For more information on IPF please see <http://coombs.anu.edu.au/ipfilter/> Each trace provides the following information: The timestamp, the hostname that the firewall process is running on (dragon), the name and process id of the IPF process (ipmon[502]), the name of the interface (fxp0), the rule group and rule number in that group which triggered IPF (0:23), whether the packet was blocked or passed and in this case blocked (b), the source IP and port, the destination IP and port, the protocol (PR tcp), the IP header length (20 bytes), the total packet length, in this case the IP header length plus the TCP header length (20 bytes) plus data (4bytes) for a total packet length of 44 bytes, and finally the TCP flags that were set (-AS)

PROBABILITY THE SOURCE ADDRESS WAS SPOOFED:

Inbound SYN/ACK packets highly suggest the potential of the source being spoofed. It can be explained as a second-order effect of a Denial of Service attack, or a probe on another site. It is known that this network did not initiate the connection with the appropriate SYN packet. It also looks highly likely that our IP address space (x.y.35.z) has been spoofed to attack someone else.

DESCRIPTION OF ATTACK :

Before we placed the OpenBSD Firewall package to our lab network, the IP network space x.y.35.0 was wide open. There is no doubt that particular hosts were compromised and put in some cracker's vulnerable host database. To date we still get the second order effects of a cracker(s) trying to access a spoofed known host on our network. The significant evidence of spoofing is that presently these hosts do not exist on our current network. Another possibility is that a cracker

maybe attempting to penetrate a firewall that allows 'established' connections to go through (hence the SYN/ACK). Open BSD Firewalls are state-full and without proper understanding of filtering rules, the firewall admin could be allowing anomalous traffic to pass through the 'force-shield'. If this is the case, then we can assume that these are crafted packets made by a tool that can bypass firewalls.

ATTACK MECHANISM :

It seems as if the attacker has a tool that probes for and invokes spoofed hosts. The attack doesn't work because the hosts do not exist and the firewall blocks this type of traffic.

CORRELATIONS :

<http://docs.rinet.ru/LomamVse/ch28/ch28.htm> discusses the spoofing attack and how it relates to this trace.

EVIDENCE OF ACTIVE TARGETING :

The attacker seems to know the network and attempts to use or abuse certain hosts on that subnet.

SEVERITY :

Criticality: PC based Firewall > 5

Lethality: Lockout due to Denial of Service > 4

System Countermeasures: OpenBSD 2.6 > 5

Network Countermeasures: Restrictive Firewall > 4

$(\text{Criticality} + \text{Lethality}) - (\text{System} + \text{Network Countermeasures}) = \text{Severity}$ $(5+4)-(5+4) = 0$ Very Low Risk...But
"If a vulnerability is never targeted, is it really a vulnerability"?

DEFENSE RECOMMENDATION :

By using packet-filtering at the department level, an enterprise can reduce the capability of attackers mapping their internal networks, scanning their hosts for vulnerabilities, and commencing exploits on those exposed hosts. An Open BSD box configured as a packet-filter can provide a 'force-shield' to block unauthorized access. Thus the recommendation is to have departmental packet filtering which blocks traffic deemed anomalous by the security analyst.

MULTIPLE CHOICE TEST QUESTION :

SYN/ACK packets without evidence of the original SYN packet are evidence of:

1. Successful second part of a trusted TCP connection establishment
2. Unsuccessful second part of a trusted TCP connection establishment
3. Potential spoofing of your IP address space
4. None of the above

The answer is (3) because state-full packet filters will show effects of the original SYN packet. If an original SYN can't be matched with a SYN/ACK then there is a high probability that your IP address space is being spoofed.

DETECT # :

Detect 4

```
July 24 17:55:24.335923 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.32,109 PR tcp len 20 40 -SF
July 24 17:55:24.486090 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.33,109 PR tcp len 20 40 -SF
July 24 17:55:24.487962 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.34,109 PR tcp len 20 40 -SF
July 24 17:55:24.595024 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.35,109 PR tcp len 20 40 -SF
July 24 17:55:24.609860 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.36,109 PR tcp len 20 40 -SF
July 24 17:55:24.677644 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.37,109 PR tcp len 20 40 -SF
July 24 17:55:24.689341 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.38,109 PR tcp len 20 40 -SF
.....
July 24 17:55:25.475923 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.77,109 PR tcp len 20 40 -SF
July 24 17:55:25.476768 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.78,109 PR tcp len 20 40 -SF
July 24 17:55:25.484572 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.79,109 PR tcp len 20 40 -SF
July 24 17:55:25.493573 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.80,109 PR tcp len 20 40 -SF
July 24 17:55:25.523802 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.81,109 PR tcp len 20 40 -SF
July 24 17:55:25.575903 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.82,109 PR tcp len 20 40 -SF
July 24 17:55:25.597834 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.83,109 PR tcp len 20 40 -SF
July 24 17:55:25.600343 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.84,109 PR tcp len 20 40 -SF
.....
July 24 17:55:26.759231 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.119,109 PR tcp len 20 40 -SF
July 24 17:55:26.773478 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.120,109 PR tcp len 20 40 -SF
July 24 17:55:26.794572 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.121,109 PR tcp len 20 40 -SF
July 24 17:55:26.827612 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.122,109 PR tcp len 20 40 -SF
July 24 17:55:26.856734 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.123,109 PR tcp len 20 40 -SF
July 24 17:55:26.863450 dragon ipmon[502] fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.124,109 PR tcp len 20 40 -SF
```

July 24 17:55:26.892361 dragon ipmon[502]	fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.125,109 PR tcp len 20 40 -SF
July 24 17:55:26.900234 dragon ipmon[502]	fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.126,109 PR tcp len 20 40 -SF
July 24 17:55:26.932175 dragon ipmon[502]	fxp0 @0:23 b 204.26.120.60,109 -> x.y.35.127,109 PR tcp len 20 40 -SF

SOURCE OF TRACE :

Protected Lab Network

DETECT WAS GENERATED BY :

IP Filter firewall package protecting a lab network. The OS is Open BSD 2.6 on an Intel Machine. For more information on IPF please see <http://coombs.anu.edu.au/ipfilter/> Each trace provides the following information: The timestamp, the hostname that the firewall process is running on (dragon), the name and process id of the IPF process (ipmon[502]), the name of the interface (fxp0), the rule group and rule number in that group which triggered IPF (0:23), whether the packet was blocked or passed and in this case blocked (b), the source IP and port, the destination IP and port, the protocol (PR tcp), the IP header length (20 bytes), the total packet length, in this case the IP header length plus the TCP header length (20 bytes) for a total packet length of 40 bytes, and finally the TCP flags that were set (-SF)

PROBABILITY THE SOURCE ADDRESS WAS SPOOFED:

The Source IP is from a corporation called Lasemaster Technologies in Minnesota. It recently changed its name to VirtualFund, Inc an investment company dedicated to the development and incubation of companies that offer Internet-based business-to-business solutions. It is highly unlikely that such a company would attack Internet a site therefore there is a very high probability that the source is spoofed.

DESCRIPTION OF ATTACK :

The attacker scanned consecutive subnets looking for a POP2 server. The SYN/FIN flag combination does not appear in normal TCP/IP traffic and is the result of a crafted packet generator or some type of automated scanning tool. FIN is what

is used to tell a system that a connection is being terminated. SYN is what is used to tell a system that a connection is being made. Attackers use SYN/FIN flags together to attempt to bypass firewalls and to evade intrusion detection systems. Although POP2 (port 109) has been largely replaced with POP3 (port 110), Many POP3 servers have a backward compatibility to POP2, presenting holes that can be exploited. The attack took approximately 2-3 seconds. The OpenBSD firewall keeps track of the state of each TCP connection and alarms on RST, SYN/ACK, SYN/FIN or FIN packets, which are detected without the active open.

ATTACK MECHANISM :

An automated tool created this scan pattern based on the short time duration of the event. The attacker then enhances his chances for a successful penetration by using SYN/FIN packets to evade detection because FIN packets may go through whereas SYN packets may get blocked. FIN packets signal connection teardown and some logging systems may not monitor the event.

CORRELATIONS :

<http://www.pulhas.org/phrack/51/P51-11.html> Phrack Magazine has an article that discusses stealth scanning. This trace appears to be an attempt to be a stealthy attacker.

EVIDENCE OF ACTIVE TARGETING :

The attacker attempts to find POP2 vulnerabilities by targeting hosts from x.y.35.32 to x.y.35.127 consecutively.

SEVERITY :

Criticality: PC based Firewall > 5
Lethality: Lockout due to Denial of Service > 4
System Countermeasures: OpenBSD 2.6 > 5

Network Countermeasures: Restrictive Firewall > 4
(Criticality + Lethality) – (System + Network Countermeasures) = Severity (5+4)-(5+4) = 0 Very Low Risk...But
“If a vulnerability is never targeted, is it really a vulnerability”?

DEFENSE RECOMMENDATION :

By using packet-filtering at the department level, an enterprise can reduce the capability of attackers mapping their internal networks, scanning their hosts for vulnerabilities, and commencing exploits on those exposed hosts. An Open BSD box configured as a packet-filter can provide a ‘force-shield’ to block unauthorized access. Thus the recommendation is to have departmental packet filtering which blocks traffic deemed anomalous by the security analyst.

MULTIPLE CHOICE TEST QUESTION26 :

SYN/FIN packets are examples of what condition?

1. Normal TCP transactions
2. Second part of a successful TCP connection establishment
3. Crafted packet
4. None of the above

The answer is (3) because SYN/FIN packets are not normal in TCP transactions and these packets are the result of a tool that crafts such packets.

DETECT # :

Detect 5

...From Morocco, ONPT Noeud Internet:

```
05/05 11:56:28.436041 194.204.233.248.2838 > .edu.1243: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 8016)
05/05 11:56:28.438783 194.204.233.248.2839 > .edu.12345: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 8272)
05/05 11:56:31.364639 194.204.233.248.2839 > .edu.12345: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 11088)
05/05 11:56:31.371635 194.204.233.248.2838 > .edu.1243: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 11344)
05/05 11:56:37.428781 194.204.233.248.2839 > .edu.12345: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 17488)
05/05 11:56:37.429534 194.204.233.248.2838 > .edu.1243: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 17744)
05/05 11:56:49.369511 194.204.233.248.2839 > .edu.12345: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 21328)
05/05 11:56:49.376024 194.204.233.248.2838 > .edu.1243: S 19040469:19040469(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 114, id 21584)
```

SOURCE OF TRACE :

The GIAC web page (<http://www.sans.org./giac.htm>) The trace is specifically from the May 8th detects:
<http://www.sans.org/y2k/050800.htm>

DETECT WAS GENERATED BY :

The trace seems to have been generated by a TCPDUMP filter that looked for SYN packets' going to ports 1243 and 12345 on .edu.

PROBABILITY THE SOURCE ADDRESS WAS SPOOFED:

There is a strong possibility that the address was spoofed, but not enough information exists to be absolutely sure. Morocco is part of fourth fifth of countries that spew malicious traffic.

DESCRIPTION OF ATTACK :

The attack is a Trojan port probe of two known Trojan ports, looking to invoke the Trojan, once it awakens through the probe, i.e. the SYN packet sent to the port.

ATTACK MECHANISM :

A tool which is capable of initiating a Trojan probe looks for SubSeven (port 1243), and Netbus or GabanBus (port 12345). It appears that the tool looks for each port in pairs. Another piece of evidence, which shows that the tool is automated, is the non-changing sequence numbers. Normal TCP/IP transactions have random sequence numbers. The time that the traces occurred suggest an automated tool as well.

CORRELATIONS :

<http://www.robertgraham.com/pubs/firewall-seen.html> discusses what a Trojan probe is as well as what SubSeven is. Also see <http://www.commodon.com/threat/threat-ports.htm>

EVIDENCE OF ACTIVE TARGETING :

Source IP 194.204.233.248 is actively targeting destination IP .edu for potential Trojans.

SEVERITY :

Criticality: Unknown since this is not a trace generated by my network; so the assumption is that this host is a UNIX desktop system > 2

Lethality Attacker can gain root access over the net > 5

System Countermeasures: Unknown; so the assumption is that in today's world this system is a Modern OS > 4

Network Countermeasures: Unknown; so the assumption is that the box is being successfully probed behind the firewall thus the firewall has let attacker through > 2

(Criticality + Lethality) – (System + Network Countermeasures) = Severity (2+5)-(4+2) = 1 Very Low Risk...But the admin should be concerned that someone is able to probe his box through a firewall...unless the box itself is a firewall or IDS

DEFENSE RECOMMENDATION :

A concern here is what .edu actually is; is it a firewall, an IDS, a router, or a desktop?. There is a strong possibility that this is a desktop machine that had TCPDUMP running on it in order to determine what traffic was being generated from it. If the machine were a firewall, packet filter, or IDS...it would have different logs to analyze. Thus the administrator needs to implement a strong firewall that will block such requests from his internal network.

MULTIPLE CHOICE TEST QUESTION :

SYN packets with static sequence numbers are usually evidence of

1. Crafted packets from an automated tool
2. Standard TCP/IP stack operations
3. Fragments will be coming down the wire soon
4. None of the above.

The answer is 1 > static sequence numbers are an anomalous condition, which should be further investigated.

Assignment 2: Evaluate an Attack

Vulnerability Scanners check a network for holes that could lead to security breaches. A good network test is to use such a scanner against your firewall. The results would either confirm or deny whether the firewall is misconfigured or has services open for exploitation. These scanners suggest fixes for problems, such as the latest OS patches, but do not take any action themselves.

If a Security Auditor can check the strength of his firewall, so can the cracker. A Vulnerability Scanner can provide so much reconnaissance data to a would-be cracker. Usually if he is able to find the holes in the firewall, he will have access to the internal network and can also scan servers for vulnerabilities, and launch the appropriate exploit.

The coolest tool to use for this analysis was Security Auditor Research Assistant (SARA) <http://www-arc.com/sara/>. SARA compiles on UNIX platforms and uses PERL plus a slick web interface to the browser of your choice. It is also SANS Certified, updated twice a month, and based on the SATAN <http://www.porcupine.org/satan/> model. But the most important aspect of SARA is that it scans for the 10 most critical Internet security threats as defined by the SANS Institute.

The following set of traces are the output of a PERL script that summarizes all blocked packets to an OpenBSD IPF Firewall package. X.y.35.19 runs the SARA tool against the IPF firewall x.y.35.17. SARA sends out SYN packets on practically all well-known ports, in a rather unique way. More analysis is required to identify this scan pattern as evidence of SARA: It probes port 79/tcp, finger first, then port 111/tcp the portmapper. Finally it scans ports 1-70/tcp, 800-1014/tcp, 365-995/tcp, 900-995/tcp, and finally 1006-1007/tcp. At the very end it begins a udp scan, but the tool realized that all well-known service ports were closed so it aborted the udp scan. Also to note is that SARA did this scan in less than 4 seconds.

```
Created new entry:      Aug 15 10:51:19  x.y.35.19  1056  x.y.35.17  79    tcp    -S
Created new entry:      Aug 15 10:51:40  x.y.35.19  705   x.y.35.17  111   tcp    -S
Created new entry:      Aug 15 10:52:00  x.y.35.19  1057  x.y.35.17  1     tcp    -S
Created new entry:      Aug 15 10:52:00  x.y.35.19  1058  x.y.35.17  2     tcp    -S
Created new entry:      Aug 15 10:52:00  x.y.35.19  1059  x.y.35.17  3     tcp    -S
Created new entry:      Aug 15 10:52:00  x.y.35.19  1060  x.y.35.17  4     tcp    -S
```

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 21

Gilbert S. Green

Created new entry:	Aug 15 10:52:00	x.y.35.19	1061	x.y.35.17	5	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1062	x.y.35.17	6	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1063	x.y.35.17	7	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1064	x.y.35.17	8	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1065	x.y.35.17	9	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1066	x.y.35.17	10	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1067	x.y.35.17	11	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1068	x.y.35.17	12	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1069	x.y.35.17	13	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1070	x.y.35.17	14	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1071	x.y.35.17	15	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1072	x.y.35.17	16	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1073	x.y.35.17	17	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1074	x.y.35.17	18	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1075	x.y.35.17	19	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1076	x.y.35.17	20	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1077	x.y.35.17	21	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1078	x.y.35.17	22	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1079	x.y.35.17	23	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1080	x.y.35.17	24	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1081	x.y.35.17	25	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1082	x.y.35.17	26	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1083	x.y.35.17	27	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1084	x.y.35.17	28	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1085	x.y.35.17	29	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1086	x.y.35.17	30	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1087	x.y.35.17	31	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1088	x.y.35.17	32	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1089	x.y.35.17	33	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1090	x.y.35.17	34	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1091	x.y.35.17	35	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1092	x.y.35.17	36	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1093	x.y.35.17	37	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1094	x.y.35.17	38	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1095	x.y.35.17	39	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1096	x.y.35.17	40	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1097	x.y.35.17	41	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 22

Gilbert S. Green

Created new entry:	Aug 15 10:52:00	x.y.35.19	1098	x.y.35.17	42	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1099	x.y.35.17	43	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1100	x.y.35.17	44	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1101	x.y.35.17	45	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1102	x.y.35.17	46	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1103	x.y.35.17	47	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1104	x.y.35.17	48	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1105	x.y.35.17	49	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1106	x.y.35.17	50	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1107	x.y.35.17	51	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1108	x.y.35.17	52	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1109	x.y.35.17	53	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1110	x.y.35.17	54	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1111	x.y.35.17	55	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1112	x.y.35.17	56	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1113	x.y.35.17	57	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1114	x.y.35.17	58	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1115	x.y.35.17	59	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1116	x.y.35.17	60	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1117	x.y.35.17	61	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1118	x.y.35.17	62	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1119	x.y.35.17	63	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1120	x.y.35.17	64	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1121	x.y.35.17	65	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1122	x.y.35.17	66	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1123	x.y.35.17	67	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1124	x.y.35.17	68	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1125	x.y.35.17	69	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1126	x.y.35.17	70	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1856	x.y.35.17	800	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1857	x.y.35.17	801	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1858	x.y.35.17	802	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1859	x.y.35.17	803	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1865	x.y.35.17	809	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1866	x.y.35.17	810	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1867	x.y.35.17	811	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1868	x.y.35.17	812	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 23

Gilbert S. Green

Created new entry:	Aug 15 10:52:00	x.y.35.19	1871	x.y.35.17	815	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1872	x.y.35.17	816	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1873	x.y.35.17	817	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1874	x.y.35.17	818	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1879	x.y.35.17	823	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1880	x.y.35.17	824	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1881	x.y.35.17	825	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1882	x.y.35.17	826	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1887	x.y.35.17	831	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1888	x.y.35.17	832	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1889	x.y.35.17	833	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1890	x.y.35.17	834	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1895	x.y.35.17	839	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1896	x.y.35.17	840	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1897	x.y.35.17	841	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1898	x.y.35.17	842	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1903	x.y.35.17	847	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1904	x.y.35.17	848	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1905	x.y.35.17	849	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1906	x.y.35.17	850	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1910	x.y.35.17	854	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1911	x.y.35.17	855	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1912	x.y.35.17	856	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1913	x.y.35.17	857	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1922	x.y.35.17	866	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1923	x.y.35.17	867	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1924	x.y.35.17	868	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1925	x.y.35.17	869	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1929	x.y.35.17	873	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1930	x.y.35.17	874	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1931	x.y.35.17	875	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1932	x.y.35.17	876	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1937	x.y.35.17	881	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1938	x.y.35.17	882	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1939	x.y.35.17	883	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1940	x.y.35.17	884	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1943	x.y.35.17	887	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 24

Gilbert S. Green

Created new entry:	Aug 15 10:52:00	x.y.35.19	1944	x.y.35.17	888	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1945	x.y.35.17	889	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1946	x.y.35.17	890	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1951	x.y.35.17	895	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1952	x.y.35.17	896	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1953	x.y.35.17	897	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1954	x.y.35.17	898	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1957	x.y.35.17	901	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1958	x.y.35.17	902	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1959	x.y.35.17	903	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1960	x.y.35.17	904	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1964	x.y.35.17	908	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1965	x.y.35.17	909	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1966	x.y.35.17	910	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1967	x.y.35.17	911	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1969	x.y.35.17	913	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1970	x.y.35.17	914	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1971	x.y.35.17	915	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1972	x.y.35.17	916	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1976	x.y.35.17	920	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1977	x.y.35.17	921	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1978	x.y.35.17	922	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1979	x.y.35.17	923	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1983	x.y.35.17	927	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1984	x.y.35.17	928	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1985	x.y.35.17	929	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1986	x.y.35.17	930	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1990	x.y.35.17	934	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1991	x.y.35.17	935	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1992	x.y.35.17	936	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1993	x.y.35.17	937	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1996	x.y.35.17	940	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1997	x.y.35.17	941	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1998	x.y.35.17	942	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	1999	x.y.35.17	943	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2003	x.y.35.17	947	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2004	x.y.35.17	948	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 25

Gilbert S. Green

Created new entry:	Aug 15 10:52:00	x.y.35.19	2005	x.y.35.17	949	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2006	x.y.35.17	950	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2009	x.y.35.17	953	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2010	x.y.35.17	954	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2011	x.y.35.17	955	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2012	x.y.35.17	956	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2016	x.y.35.17	960	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2017	x.y.35.17	961	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2018	x.y.35.17	962	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2019	x.y.35.17	963	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2023	x.y.35.17	967	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2024	x.y.35.17	968	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2025	x.y.35.17	969	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2026	x.y.35.17	970	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2029	x.y.35.17	973	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2030	x.y.35.17	974	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2031	x.y.35.17	975	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2032	x.y.35.17	976	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2034	x.y.35.17	978	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2035	x.y.35.17	979	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2036	x.y.35.17	980	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2037	x.y.35.17	981	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2040	x.y.35.17	984	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2041	x.y.35.17	985	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2042	x.y.35.17	986	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2043	x.y.35.17	987	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2046	x.y.35.17	990	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2047	x.y.35.17	991	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2048	x.y.35.17	992	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2049	x.y.35.17	993	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2052	x.y.35.17	996	tcp	-S
Created new entry:	Aug 15 10:52:00	x.y.35.19	2053	x.y.35.17	997	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2054	x.y.35.17	998	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2055	x.y.35.17	999	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2058	x.y.35.17	1002	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2059	x.y.35.17	1003	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2060	x.y.35.17	1004	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 26

Gilbert S. Green

Created new entry:	Aug 15 10:52:01	x.y.35.19	2061	x.y.35.17	1005	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2064	x.y.35.17	1008	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2065	x.y.35.17	1009	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2066	x.y.35.17	1010	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2067	x.y.35.17	1011	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2069	x.y.35.17	1013	tcp	-S
Created new entry:	Aug 15 10:52:01	x.y.35.19	2070	x.y.35.17	1014	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1421	x.y.35.17	365	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1422	x.y.35.17	366	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1423	x.y.35.17	367	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1424	x.y.35.17	368	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1425	x.y.35.17	369	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1426	x.y.35.17	370	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1427	x.y.35.17	371	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1428	x.y.35.17	372	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1449	x.y.35.17	393	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1450	x.y.35.17	394	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1451	x.y.35.17	395	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1452	x.y.35.17	396	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1472	x.y.35.17	416	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1473	x.y.35.17	417	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1474	x.y.35.17	418	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1475	x.y.35.17	419	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1491	x.y.35.17	435	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1492	x.y.35.17	436	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1493	x.y.35.17	437	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1494	x.y.35.17	438	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1498	x.y.35.17	442	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1499	x.y.35.17	443	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1500	x.y.35.17	444	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1501	x.y.35.17	445	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1523	x.y.35.17	467	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1524	x.y.35.17	468	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1525	x.y.35.17	469	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1526	x.y.35.17	470	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1546	x.y.35.17	490	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1547	x.y.35.17	491	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 27

Gilbert S. Green

Created new entry:	Aug 15 10:53:33	x.y.35.19	1548	x.y.35.17	492	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1549	x.y.35.17	493	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1569	x.y.35.17	513	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1570	x.y.35.17	514	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1571	x.y.35.17	515	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1572	x.y.35.17	516	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1573	x.y.35.17	517	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1574	x.y.35.17	518	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1575	x.y.35.17	519	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1576	x.y.35.17	520	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1589	x.y.35.17	533	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1590	x.y.35.17	534	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1591	x.y.35.17	535	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1592	x.y.35.17	536	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1610	x.y.35.17	554	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1611	x.y.35.17	555	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1612	x.y.35.17	556	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1613	x.y.35.17	557	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1614	x.y.35.17	558	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1615	x.y.35.17	559	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1648	x.y.35.17	592	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1649	x.y.35.17	593	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1650	x.y.35.17	594	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1651	x.y.35.17	595	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1652	x.y.35.17	596	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1653	x.y.35.17	597	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1663	x.y.35.17	607	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1664	x.y.35.17	608	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1665	x.y.35.17	609	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1666	x.y.35.17	610	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1681	x.y.35.17	625	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1682	x.y.35.17	626	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1683	x.y.35.17	627	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1684	x.y.35.17	628	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1685	x.y.35.17	629	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1686	x.y.35.17	630	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1687	x.y.35.17	631	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 28

Gilbert S. Green

Created new entry:	Aug 15 10:53:33	x.y.35.19	1688	x.y.35.17	632	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1695	x.y.35.17	639	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1696	x.y.35.17	640	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1697	x.y.35.17	641	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1698	x.y.35.17	642	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1712	x.y.35.17	656	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1713	x.y.35.17	657	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1714	x.y.35.17	658	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1715	x.y.35.17	659	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1716	x.y.35.17	660	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1717	x.y.35.17	661	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1718	x.y.35.17	662	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1719	x.y.35.17	663	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1728	x.y.35.17	672	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1729	x.y.35.17	673	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1730	x.y.35.17	674	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1731	x.y.35.17	675	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1744	x.y.35.17	688	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1745	x.y.35.17	689	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1746	x.y.35.17	690	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1747	x.y.35.17	691	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1748	x.y.35.17	692	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1749	x.y.35.17	693	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1750	x.y.35.17	694	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1751	x.y.35.17	695	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1759	x.y.35.17	703	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1760	x.y.35.17	704	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1761	x.y.35.17	705	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1762	x.y.35.17	706	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1774	x.y.35.17	718	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1775	x.y.35.17	719	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1776	x.y.35.17	720	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1777	x.y.35.17	721	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1778	x.y.35.17	722	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1779	x.y.35.17	723	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1780	x.y.35.17	724	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1781	x.y.35.17	725	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 29

Gilbert S. Green

Created new entry:	Aug 15 10:53:33	x.y.35.19	1788	x.y.35.17	732	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1789	x.y.35.17	733	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1790	x.y.35.17	734	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1805	x.y.35.17	749	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1806	x.y.35.17	750	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1807	x.y.35.17	751	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1808	x.y.35.17	752	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1809	x.y.35.17	753	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1810	x.y.35.17	754	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1812	x.y.35.17	756	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1813	x.y.35.17	757	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1869	x.y.35.17	813	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1870	x.y.35.17	814	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1876	x.y.35.17	820	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1877	x.y.35.17	821	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1878	x.y.35.17	822	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1891	x.y.35.17	835	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1892	x.y.35.17	836	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1893	x.y.35.17	837	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1894	x.y.35.17	838	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1899	x.y.35.17	843	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1900	x.y.35.17	844	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1901	x.y.35.17	845	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1902	x.y.35.17	846	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1926	x.y.35.17	870	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1927	x.y.35.17	871	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1928	x.y.35.17	872	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1933	x.y.35.17	877	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1934	x.y.35.17	878	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1935	x.y.35.17	879	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1936	x.y.35.17	880	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1947	x.y.35.17	891	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1948	x.y.35.17	892	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1949	x.y.35.17	893	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1950	x.y.35.17	894	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1955	x.y.35.17	899	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1956	x.y.35.17	900	tcp	-S

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 30

Gilbert S. Green

Created new entry:	Aug 15 10:53:33	x.y.35.19	1968	x.y.35.17	912	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1987	x.y.35.17	931	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1994	x.y.35.17	938	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	1995	x.y.35.17	939	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2000	x.y.35.17	944	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2001	x.y.35.17	945	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2007	x.y.35.17	951	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2014	x.y.35.17	958	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2015	x.y.35.17	959	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2027	x.y.35.17	971	tcp	-S
Created new entry:	Aug 15 10:53:33	x.y.35.19	2033	x.y.35.17	977	tcp	-S
Created new entry:	Aug 15 10:53:34	x.y.35.19	2044	x.y.35.17	988	tcp	-S
Created new entry:	Aug 15 10:53:34	x.y.35.19	2045	x.y.35.17	989	tcp	-S
Created new entry:	Aug 15 10:53:34	x.y.35.19	2051	x.y.35.17	995	tcp	-S
Created new entry:	Aug 15 10:53:34	x.y.35.19	2062	x.y.35.17	1006	tcp	-S
Created new entry:	Aug 15 10:53:34	x.y.35.19	2063	x.y.35.17	1007	tcp	-S
Created new entry:	Aug 15 10:55:01	x.y.35.19	1024	x.y.35.17	1	udp	

The following report is a cut-and-pasted version of a SARA report. It shows the vulnerabilities that exist in the scanned host or network. This report shows the vulnerabilities that were supposedly evident on our lab firewall.

Security Auditor's Research Assistant (SARA) Professional (PRO) Report Writer
Back to the SARA start page | Back to SARA Reporting and Analysis
place date here
place document number here

SARA Scan Results of Customer Name

INTRODUCTION

Advanced Research Corporation ® was tasked to perform a Security Auditor's Research Assistant (SARA) security scan on hosts on the Customer Name sub-nets.

The SARA scan was performed to identify potential security vulnerabilities in the Customer Name sub-domain.

DISCUSSION

SARA is a third generation security analysis tool that analyzes network-based services on the target computers. SARA classifies a detected service in one of four categories:

Green: Services found that were not exploitable

None: No services or vulnerabilities

Red: Services with potentially severe exploits (account compromise)

Yellow: Services with potentially serious exploits found (data compromise)

Brown: Possible security problems.

Figure 1 summarizes this scan by color where the Green bar indicates hosts with no detected vulnerabilities. None indicates hosts with no services. The Red bar indicates hosts that have one or more red vulnerabilities. The Yellow bar indicates hosts that have one or more yellow vulnerabilities (but no red). And the Brown bar indicates hosts that have one or more brown problems (but no red or yellow)

Green

0

None

1 [Notice that one host (the firewall) has no services; The actual report would put

Red Colored bar-graphs here.]
0
Yellow
0
Brown
0

Figure 1 Host Summary by Color

The SARA scan results are distributed as three appendices to this paper:

- Appendix A: Sub-net tables depicting hosts, host-types, and vulnerability counts.
- Appendix B: Details on the hosts reported
- Appendix C: Description of the vulnerabilities

Appendices A through C are hyper-linked to assist the reader in navigating through this report. The report includes information on all non-Windows hosts that have one or more vulnerabilities. In addition, Windows hosts that have Red and/or Yellow vulnerabilities are also included.

RECOMMENDATION

Notice there are no recommendations

The identified hosts should be analyzed immediately.

Notice there are no hosts that were identified

Appendix A
SARA Scan Summary
Host Name x.y.35.17
IP Address x.y.35.17
Host Type Unknown Type
Green 0
Red 0
Yellow 0

Brown 0

Table 1 Hosts on Sub-net x.y.35

Appendix B
SARA Scan Details

Host: x.y.35.17
General host information:
Host type: unknown type
Subnet x.y.35

Vulnerability information:

Notice no vulnerability data shown either

Appendix C
SARA Vulnerability Tutorials

According to the SARA report our firewall has no services running on it at all. Therefore this firewall, is industrial strength, however, if the IPF rule-sets are not properly created, the firewall admin could be opening his network up for crackers to wreak havoc upon.

Assignment 3 : "Analyze This"

The scans to be shown in this analysis are the Snort Detects from <http://www.sans.org/giactc/snort/SnortS6.txt>. After August 15, 2000, they will no longer be available for public analysis. Therefore detects analyzed in this portion of the practical have been cut and pasted from the original document. The scan report shows the detects sensed by a Snort IDS within a 24 hour period. These scans give a very high level view of security events. Therefore, without correlation, some Analyses could appear to be incomplete. Many of the ports were researched from <http://advice.networkice.com/advice/Exploits/Ports/>. This site has a handy hyper-linked source of ports.

Trace 1

```
Jun 6 00:56:01 MY.NET.1.3: 53 -> MY.NET.101.89: 38487 UDP
Jun 6 08:37:43 MY.NET.1.3:53 -> MY.NET.101.89:40045 UDP
Jun 6 08:39:33 MY.NET.1.3:53 -> MY.NET.101.89:40087 UDP
Jun 6 09:16:43 MY.NET.1.3:53 -> MY.NET.101.89:40368 UDP
Jun 6 09:19:53 MY.NET.1.3:53 -> MY.NET.101.89:40409 UDP
Jun 6 10:22:58 MY.NET.1.3:53 -> MY.NET.101.89:41094 UDP
Jun 6 10:23:01 MY.NET.1.3:53 -> MY.NET.101.89:41111 UDP
Jun 6 10:56:45 MY.NET.1.3:53 -> MY.NET.101.89:41430 UDP
Jun 6 11:06:20 MY.NET.1.3:53 -> MY.NET.101.89:41577 UDP
Jun 6 11:13:51 MY.NET.1.3:53 -> MY.NET.101.89:41709 UDP
Jun 6 12:24:42 MY.NET.1.3:53 -> MY.NET.101.89:42731 UDP
Jun 6 13:05:51 MY.NET.1.3:53 -> MY.NET.101.89:43260 UDP
Jun 6 15:31:54 MY.NET.1.3:53 -> MY.NET.101.89:44512 UDP
Jun 6 18:31:34 MY.NET.1.3:53 -> MY.NET.101.89:45737 UDP
Jun 6 18:32:44 MY.NET.1.3:53 -> MY.NET.101.89:45765 UDP
Jun 6 20:09:35 MY.NET.1.3:53 -> MY.NET.101.89:46212 UDP
Jun 6 20:11:04 MY.NET.1.3:53 -> MY.NET.101.89:46237 UDP
Jun 6 20:21:43 MY.NET.1.3:53 -> MY.NET.101.89:46289 UDP
Jun 6 23:19:33 MY.NET.1.3:53 -> MY.NET.101.89:47079 UDP
```

This activity, which occurs throughout the day, appears to be the domain's DNS server resolving a name for host MY.NET.101.89, or querying the host itself. The well-known port 53 attempts to connect to ephemeral ports 38487-47079, which appears normal. The IDS seems to be designed to log any DNS server activity within the domain. The packets are UDP, thus within the 512-byte size, and which also appears normal.

Trace 2

Jun 6 00:48:46 202.163.97.236:2652 -> MY.NET.60.11:1243 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2653 -> MY.NET.60.11:31337 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2659 -> MY.NET.60.11:3902 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2662 -> MY.NET.60.11:9191 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2664 -> MY.NET.60.11:34123 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2666 -> MY.NET.60.11:5555 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2667 -> MY.NET.60.11:28983 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2669 -> MY.NET.60.11:9009 SYN **S*****
Jun 6 00:48:46 202.163.97.236:2670 -> MY.NET.60.11:6671 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2655 -> MY.NET.60.11:21554 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2652 -> MY.NET.60.11:1243 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2653 -> MY.NET.60.11:31337 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2663 -> MY.NET.60.11:19799 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2656 -> MY.NET.60.11:6400 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2661 -> MY.NET.60.11:3333 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2671 -> MY.NET.60.11:1027 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2668 -> MY.NET.60.11:6000 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2666 -> MY.NET.60.11:5555 SYN **S*****
Jun 6 00:48:49 202.163.97.236:2670 -> MY.NET.60.11:6671 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2657 -> MY.NET.60.11:27374 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2665 -> MY.NET.60.11:14500 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2662 -> MY.NET.60.11:9191 SYN **S*****
Jun 6 00:48:50 202.163.97.236:2658 -> MY.NET.60.11:9732 SYN **S*****
Jun 6 00:48:53 202.163.97.236:2668 -> MY.NET.60.11:6000 SYN **S*****
Jun 6 00:48:53 202.163.97.236:2667 -> MY.NET.60.11:28983 SYN **S*****
Jun 6 00:48:53 202.163.97.236:2661 -> MY.NET.60.11:3333 SYN **S*****
Jun 6 00:48:53 202.163.97.236:2658 -> MY.NET.60.11:9732 SYN **S*****

This attack is a hunt for Trojans. MY.NET.60.11 is targeted by 202.163.97.236. The source is from Pakistan and is probably spoofed. Trojans identified are SubSeven (1243), Back Orifice (31337), Napster or ServeMe (5555), The Thing (6000,6400), SubSeven (27374). The Attacker is scanning MY.NET.60.11 for potential compromise. If the IDS is outside of the firewall, then these scans have low priority. If there is no firewall, the attacker can implant a Trojan.

Trace 3

```
Jun 6 01:38:50 202.163.97.236:3602 -> MY.NET.60.11:31337 SYN **S*****
Jun 6 01:38:50 202.163.97.236:3612 -> MY.NET.60.11:19799 SYN **S*****
Jun 6 01:38:51 202.163.97.236:3619 -> MY.NET.60.11:6671 SYN **S*****
Jun 6 01:38:51 202.163.97.236:3616 -> MY.NET.60.11:28983 SYN **S*****
Jun 6 01:38:51 202.163.97.236:3603 -> MY.NET.60.11:6670 SYN **S*****
Jun 6 01:38:51 202.163.97.236:3609 -> MY.NET.60.11:5180 SYN **S*****
Jun 6 01:38:51 202.163.97.236:3613 -> MY.NET.60.11:34123 SYN **S*****
Jun 6 01:38:52 202.163.97.236:3620 -> MY.NET.60.11:1027 SYN **S*****
Jun 6 01:38:52 202.163.97.236:3600 -> MY.NET.60.11:12345 SYN **S*****
Jun 6 01:38:54 202.163.97.236:3601 -> MY.NET.60.11:1243 SYN **S*****
```

This is continuation of the previous attack shown on trace 2, just at a later time.

Trace 4

```
Jun 6 03:05:38 198.186.203.18:20 -> MY.NET.253.105:14220 SYN **S*****
Jun 6 03:06:34 198.186.203.18:20 -> MY.NET.253.105:14342 SYN **S*****
Jun 6 03:07:55 198.186.203.18:20 -> MY.NET.253.105:14500 SYN **S*****
```

The source IP, from San Jose CA, attempts to SYN-flood MY.NET.253.105 with 141 SYN packets delivered within approximately two seconds. The above trace has been shortened for brevity but includes the first detect, a middle detect and the last detect. The trace also shows that the FTP command channel is being used to send the SYN packets to consecutive ports on MY.NET.253.105.

Trace 5

```
Jun 6 15:22:19 198.86.17.38:2048 -> MY.NET.6.7:29907 NULL *****  
Jun 6 15:22:24 198.86.17.38:4028 -> MY.NET.6.7:53 FIN ***F****
```

A source IP from North Carolina sends two curious packets to MY.NET.6.7, a null packet and a FIN packet. FIN packets and Null packets sent to a port as stimulus will invoke a RST from closed ports and no response from open ports, because these packets are dropped. FIN, NULL and XMAS probes are designed to evade firewalls and intrusion detection systems. For the two detects show an attempt to be extremely stealthy, while probing for information. Notice that the prober attempts to see if DNS service is available on MY.NET.6.7.

Trace 6

```
Jun 6 03:21:36 194.106.96.11:47312 -> MY.NET.221.186:21 SYN **S*****  
Jun 6 03:21:37 194.106.96.11:47312 -> MY.NET.221.186:21 NULL *****
```

The source IP is from Estonia and sends two targeted probes to MY.NET.221.186 on port 21, the FTP data channel port. This is assumed to be a low slow scan using the SYN packet as a first level stealth probe to the destination IP. Immediately a null scan is attempted. Null scans are super stealth techniques designed to evade intrusion detection systems and firewalls.

Trace 7

```
Jun 6 04:29:43 61.130.138.205:40914 -> MY.NET.253.112:443 INVALIDACK ***FRPA*
Jun 6 04:37:03 61.130.138.205:41122 -> MY.NET.253.112:443 INVALIDACK ***FRPA*
Jun 6 04:47:27 61.130.138.205:41583 -> MY.NET.253.112:443 INVALIDACK ***FRPA*
Jun 6 04:51:11 61.130.138.205:41685 -> MY.NET.253.112:443 INVALIDACK ***FRPA*
Jun 6 19:07:56 172.135.125.187:14926 -> MY.NET.253.112:443 NULL *****
```

Two different source IP addresses scan the same host at different times, each with a different scan technique. Destination port 443 is Secure HTTP or https. This is http over ssl. The first four packets are oddball and do not comply with standard TCP traffic, therefore it is anomalous. The last packet is a null scan, an attempt to be stealthy. The first source is from China and the second source is from AOL. This could be evidence of a coordinated attack.

Trace 8

```
Jun 6 06:58:22 193.195.1.5:27950 -> MY.NET.20.10:27960 UNKNOWN 2***R*A* RESERVEDBITS
Jun 6 07:01:20 193.195.1.5:0 -> MY.NET.20.10:56184 NOACK 2*SFRP** RESERVEDBITS
Jun 6 07:01:49 193.195.1.5:53 -> MY.NET.20.10:1116 FIN ***F****
Jun 6 07:58:10 193.238.203.80:4687 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:44:48 193.238.218.2:6801 -> MY.NET.20.10:7275 INVALIDACK 2***R*AU RESERVEDBITS
Jun 6 11:49:03 193.238.139.35:27015 -> MY.NET.20.10:27005 SPAU 2*S**PAU RESERVEDBITS
```

```
Jun 6 08:48:05 195.11.133.60:4687 -> MY.NET.20.10:53 FIN ***F****
Jun 6 08:51:04 195.11.243.24:7766 -> MY.NET.20.10:49154 UNKNOWN 21*F*PA* RESERVEDBITS
Jun 6 08:52:33 195.11.243.24:1141 -> MY.NET.20.10:1173 INVALIDACK ***FR*A*
Jun 6 08:54:40 195.11.133.60:137 -> MY.NET.20.10:137 FIN ***F****
Jun 6 11:32:07 195.11.122.155:53 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:35:00 195.11.82.54:1916 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:35:45 195.11.122.155:4471 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:49:13 193.238.139.35:4606 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:50:19 193.238.139.35:27035 -> MY.NET.20.10:27005 VECNA ***F***U
```

4 scans from the four different sources to MY.NET.20.10. The first scan sends an oddball packet; RST/ACK can be common; but one with reserved bits set is unusual. The second scan shows a port 0 source scan; this is anomalous and needs more correlation. The third scan from a DNS service port shows a FIN scan, an attempt to be stealthy, and extract information about the destination. The last 3 scans are from 3 different hosts and appear to be a coordinated probe. Another possibility is that the source IP addresses belong to demon-net. It has been said that this network has misconfigured routers on it.

Trace 9

```
Jun 6 07:05:34 24.23.45.19:195 -> MY.NET.6.7:1490 NOACK 2**FR*** RESERVEDBITS
Jun 6 07:05:32 24.23.45.19:1490 -> MY.NET.6.7:8554 NOACK 2**FR*** RESERVEDBITS
Jun 6 07:06:11 24.23.45.19:1490 -> MY.NET.6.7:8554 NOACK 2**FR*** RESERVEDBITS
Jun 6 07:08:01 24.23.45.19:1492 -> MY.NET.6.7:8581 INVALIDACK **SFRPA*
Jun 6 07:08:39 24.23.45.19:1492 -> MY.NET.6.7:8581 INVALIDACK **SFRPA*
Jun 6 07:08:41 24.23.45.19:195 -> MY.NET.6.7:1492 INVALIDACK **SFRPA*
Jun 6 07:08:45 24.23.45.19:1492 -> MY.NET.6.7:8581 INVALIDACK **SFRPA*
Jun 6 07:09:23 24.23.45.19:195 -> MY.NET.6.7:1495 INVALIDACK 2*SFRPA* RESERVEDBITS
Jun 6 07:09:29 24.23.45.19:1495 -> MY.NET.6.7:8590 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 07:09:33 24.23.45.19:1495 -> MY.NET.6.7:8590 INVALIDACK 2*SFRPA* RESERVEDBITS
```



```
Jun 6 07:09:39 24.23.45.19:1495 -> MY.NET.6.7:8590 INVALIDACK 2*SFRPA* RESERVEDBITS
Jun 6 07:09:44 24.23.45.19:195 -> MY.NET.6.7:1495 INVALIDACK 2*SFRPA* RESERVEDBITS
Jun 6 07:09:45 24.23.45.19:1495 -> MY.NET.6.7:8590 SYNFIN 21SF**** RESERVEDBITS
Jun 6 07:09:52 24.23.45.19:1495 -> MY.NET.6.7:8590 INVALIDACK 2*SFRPA* RESERVEDBITS
Jun 6 07:09:52 24.23.45.19:1495 -> MY.NET.6.7:8590 SYNFIN 21SF**** RESERVEDBITS
Jun 6 07:10:39 24.23.45.19:1496 -> MY.NET.6.7:8599 NOACK *1**RP** RESERVEDBITS
Jun 6 07:10:40 24.23.45.19:203 -> MY.NET.6.7:1496 NOACK *1**RP** RESERVEDBITS
Jun 6 07:11:12 24.23.45.19:1497 -> MY.NET.6.7:8607 SYN **S*****
Jun 6 07:11:13 24.23.45.19:1497 -> MY.NET.6.7:8607 INVALIDACK *1S**PA* RESERVEDBITS
Jun 6 07:11:15 24.23.45.19:1497 -> MY.NET.6.7:8607 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 07:11:33 24.23.45.19:1497 -> MY.NET.6.7:8607 SYNFIN 21SF**** RESERVEDBITS
Jun 6 07:11:43 24.23.45.19:1497 -> MY.NET.6.7:8607 INVALIDACK *1S**PA* RESERVEDBITS
Jun 6 07:11:58 24.23.45.19:1497 -> MY.NET.6.7:8607 INVALIDACK *1S**PA* RESERVEDBITS
Jun 6 07:12:02 24.23.45.19:1497 -> MY.NET.6.7:8607 INVALIDACK *1S**PA* RESERVEDBITS
Jun 6 07:12:04 24.23.45.19:1497 -> MY.NET.6.7:8607 INVALIDACK *1S**PA* RESERVEDBITS
```

Lots of strange flag combination packets being sent to MY.NET.6.7. Seems like an attempt to coordinate the proper port-packet pair. Then Coordination seems to occur once source port 1497 and destination port 8607 are chosen. What is most notable are the last four packets with the SYN, PUSH, and ACK packets sent; this could be an attempt to set up and use a covert channel; along with the reserved bits set, this type of traffic could be tracked and monitored to best

understand it. The attacker appears to be a cable modem user because the source IP is part of the @Home network address space.

Trace 10

```
Jun 6 07:23:29 146.235.25.1:10322 -> MY.NET.253.41:25 NOACK ****RP**
```

This is a JCPenny machine sending a strange packet to MY.NET.235.41 The port being attacked is the SMTP port. Could this be a Spam server looking for a new victim to send lots of advertisements about JCPenny?

Trace 11

```
Jun 6 11:42:29 195.11.19.250:53 -> MY.NET.70.77:53 FIN ***F****
Jun 6 11:44:39 195.11.19.250:7733 -> MY.NET.70.77:4295 UNKNOWN 21*F**AU RESERVEDBITS
Jun 6 12:18:01 132.230.178.178:1316 -> MY.NET.70.77:6346 NOACK 2**FR**U RESERVEDBITS
Jun 6 12:28:07 132.230.178.178:1316 -> MY.NET.70.77:6346 VECNA ***F***U
Jun 6 15:25:30 216.161.168.75:6346 -> MY.NET.70.77:2067 FIN ***F****
```

Three different source IP addresses send MY.NET.70.77 unusual packets. This could be evidence of a coordinated reconnaissance effort. The source IP addresses are from the demon-net and Denmark.

Trace 12

```
Jun 6 09:39:25 194.70.126.10:1026 -> MY.NET.253.43:53 FIN ***F****
Jun 6 19:19:39 194.70.58.41:1031 -> MY.NET.150.46:11677 INVALIDACK **SFR*AU
```

A possibility is that the source IP addresses belong to demon-net. It has been said that this network has misconfigured routers on it, thus the strange packets. But the FIN scan to port 53 on MY.NET.253.43 appears to be an attempt to use a misconfigured network to hide in the strange traffic, and deploy the stealth scan.

GIAC Certified Intrusion Analyst (GCIA) Practical

01/16/05

12:24 AM

Page 42

Gilbert S. Green

Trace 13

Jun 6 09:45:54 213.188.8.45:2487 -> MY.NET.202.142:21 SYN **S*****
Jun 6 09:45:54 213.188.8.45:2508 -> MY.NET.205.94:21 SYN **S*****
Jun 6 09:45:55 213.188.8.45:2510 -> MY.NET.206.26:21 SYN **S*****
Jun 6 09:45:55 213.188.8.45:2489 -> MY.NET.202.234:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2492 -> MY.NET.203.118:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2513 -> MY.NET.208.106:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2528 -> MY.NET.210.86:21 SYN **S*****
Jun 6 09:45:57 213.188.8.45:2521 -> MY.NET.208.74:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2529 -> MY.NET.219.174:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2515 -> MY.NET.208.42:21 SYN **S*****
Jun 6 09:45:57 213.188.8.45:2498 -> MY.NET.203.154:21 SYN **S*****
Jun 6 09:45:58 213.188.8.45:2500 -> MY.NET.203.178:21 SYN **S*****
Jun 6 09:45:58 213.188.8.45:2536 -> MY.NET.221.78:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2529 -> MY.NET.219.174:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2528 -> MY.NET.210.86:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2506 -> MY.NET.204.190:21 SYN **S*****
Jun 6 09:46:00 213.188.8.45:2508 -> MY.NET.205.94:21 SYN **S*****
Jun 6 09:46:01 213.188.8.45:2509 -> MY.NET.206.158:21 SYN **S*****
Jun 6 09:46:01 213.188.8.45:2536 -> MY.NET.221.78:21 SYN **S*****
Jun 6 20:20:14 213.188.8.45:4901 -> MY.NET.203.118:21 SYN **S*****
Jun 6 20:20:14 213.188.8.45:4902 -> MY.NET.203.154:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4906 -> MY.NET.203.178:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4910 -> MY.NET.206.158:21 SYN **S*****
Jun 6 20:20:15 213.188.8.45:4880 -> MY.NET.201.14:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4888 -> MY.NET.201.246:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4887 -> MY.NET.201.222:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4892 -> MY.NET.202.234:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4891 -> MY.NET.202.142:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4921 -> MY.NET.208.42:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4908 -> MY.NET.205.94:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4907 -> MY.NET.204.190:21 SYN **S*****
Jun 6 20:20:19 213.188.8.45:4916 -> MY.NET.208.106:21 SYN **S*****
Jun 6 20:20:19 213.188.8.45:4915 -> MY.NET.206.26:21 SYN **S*****
Jun 6 20:20:21 213.188.8.45:4928 -> MY.NET.209.42:21 SYN **S*****

Jun 6 20:20:21 213.188.8.45:4880 -> MY.NET.201.14:21 SYN **S*****
Jun 6 20:20:22 213.188.8.45:4888 -> MY.NET.201.246:21 SYN **S*****
Jun 6 20:20:23 213.188.8.45:4959 -> MY.NET.221.78:21 SYN **S*****
Jun 6 20:20:23 213.188.8.45:4901 -> MY.NET.203.118:21 SYN **S*****
Jun 6 20:20:23 213.188.8.45:4902 -> MY.NET.203.154:21 SYN **S*****
Jun 6 20:20:25 213.188.8.45:4916 -> MY.NET.208.106:21 SYN **S*****
Jun 6 20:20:25 213.188.8.45:4915 -> MY.NET.206.26:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2529 -> MY.NET.219.174:21 SYN **S*****
Jun 6 09:45:56 213.188.8.45:2515 -> MY.NET.208.42:21 SYN **S*****
Jun 6 09:45:57 213.188.8.45:2498 -> MY.NET.203.154:21 SYN **S*****
Jun 6 09:45:58 213.188.8.45:2500 -> MY.NET.203.178:21 SYN **S*****
Jun 6 09:45:58 213.188.8.45:2536 -> MY.NET.221.78:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2529 -> MY.NET.219.174:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2528 -> MY.NET.210.86:21 SYN **S*****
Jun 6 09:45:59 213.188.8.45:2506 -> MY.NET.204.190:21 SYN **S*****
Jun 6 09:46:00 213.188.8.45:2508 -> MY.NET.205.94:21 SYN **S*****
Jun 6 09:46:01 213.188.8.45:2509 -> MY.NET.206.158:21 SYN **S*****
Jun 6 09:46:01 213.188.8.45:2536 -> MY.NET.221.78:21 SYN **S*****
Jun 6 20:20:14 213.188.8.45:4901 -> MY.NET.203.118:21 SYN **S*****
Jun 6 20:20:14 213.188.8.45:4902 -> MY.NET.203.154:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4906 -> MY.NET.203.178:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4910 -> MY.NET.206.158:21 SYN **S*****
Jun 6 20:20:15 213.188.8.45:4880 -> MY.NET.201.14:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4888 -> MY.NET.201.246:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4887 -> MY.NET.201.222:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4892 -> MY.NET.202.234:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4891 -> MY.NET.202.142:21 SYN **S*****
Jun 6 20:20:16 213.188.8.45:4921 -> MY.NET.208.42:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4908 -> MY.NET.205.94:21 SYN **S*****
Jun 6 20:20:18 213.188.8.45:4907 -> MY.NET.204.190:21 SYN **S*****
Jun 6 20:20:19 213.188.8.45:4916 -> MY.NET.208.106:21 SYN **S*****
Jun 6 20:20:19 213.188.8.45:4915 -> MY.NET.206.26:21 SYN **S*****
Jun 6 20:20:21 213.188.8.45:4928 -> MY.NET.209.42:21 SYN **S*****
Jun 6 20:20:21 213.188.8.45:4880 -> MY.NET.201.14:21 SYN **S*****
Jun 6 20:20:22 213.188.8.45:4888 -> MY.NET.201.246:21 SYN **S*****
Jun 6 20:20:23 213.188.8.45:4959 -> MY.NET.221.78:21 SYN **S*****
Jun 6 20:20:23 213.188.8.45:4901 -> MY.NET.203.118:21 SYN **S*****

Jun 6 20:20:23 213.188.8.45:4902 -> MY.NET.203.154:21 SYN **S*****
Jun 6 20:20:25 213.188.8.45:4916 -> MY.NET.208.106:21 SYN **S*****
Jun 6 20:20:25 213.188.8.45:4915 -> MY.NET.206.26:21 SYN **S*****
Jun 6 20:20:25 213.188.8.45:4921 -> MY.NET.208.42:21 SYN **S*****
Jun 6 20:20:28 213.188.8.45:4931 -> MY.NET.210.86:21 SYN **S*****
Jun 6 20:20:31 213.188.8.45:4948 -> MY.NET.219.174:21 SYN **S*****
Jun 6 20:20:33 213.188.8.45:4880 -> MY.NET.201.14:21 SYN **S*****
Jun 6 20:20:34 213.188.8.45:4888 -> MY.NET.201.246:21 SYN **S*****
Jun 6 20:20:34 213.188.8.45:4891 -> MY.NET.202.142:21 SYN **S*****
Jun 6 20:20:35 213.188.8.45:4901 -> MY.NET.203.118:21 SYN **S*****
Jun 6 20:20:35 213.188.8.45:4902 -> MY.NET.203.154:21 SYN **S*****
Jun 6 20:20:36 213.188.8.45:4908 -> MY.NET.205.94:21 SYN **S*****
Jun 6 20:20:36 213.188.8.45:4910 -> MY.NET.206.158:21 SYN **S*****
Jun 6 20:20:36 213.188.8.45:4906 -> MY.NET.203.178:21 SYN **S*****
Jun 6 20:20:36 213.188.8.45:4907 -> MY.NET.204.190:21 SYN **S*****
Jun 6 20:20:37 213.188.8.45:4916 -> MY.NET.208.106:21 SYN **S*****
Jun 6 20:20:37 213.188.8.45:4915 -> MY.NET.206.26:21 SYN **S*****
Jun 6 20:20:38 213.188.8.45:4926 -> MY.NET.208.74:21 SYN **S*****
Jun 6 20:20:39 213.188.8.45:4928 -> MY.NET.209.42:21 SYN **S*****
Jun 6 20:20:40 213.188.8.45:4931 -> MY.NET.210.86:21 SYN **S*****
Jun 6 20:20:43 213.188.8.45:4948 -> MY.NET.219.174:21 SYN **S*****
Jun 6 20:20:44 213.188.8.45:4959 -> MY.NET.221.78:21 SYN **S*****

One source IP scans various destination subnets for hosts with the FTP port 21 (data channel) open using the SYN scan. The source is from Norway.

Trace 14

Jun 6 10:53:44 194.217.242.91:27995 -> MY.NET.253.42:27960 VECNA ***F*P**
Jun 6 11:01:26 194.217.20.10:53 -> MY.NET.157.150:53 FIN ***F****

```
Jun 6 11:29:00 194.217.188.53:2048 -> MY.NET.20.10:12636 NOACK 2*S*R**U RESERVEDBITS
Jun 6 11:55:43 194.217.188.53:27035 -> MY.NET.20.10:27005 NOACK **S****U
Jun 6 11:59:45 194.217.188.53:53 -> MY.NET.20.10:53 FIN ***F****
Jun 6 11:59:54 194.217.188.53:7799 -> MY.NET.20.10:1049 SYNFIN *1SF**** RESERVEDBITS
Jun 6 12:00:03 194.217.188.53:27998 -> MY.NET.20.10:62324 SYNFIN **SF****
Jun 6 13:20:04 194.217.188.53:27055 -> MY.NET.20.10:27005 INVALIDACK **S**PA*
Jun 6 14:57:23 195.173.133.162:137 -> MY.NET.20.10:137 FIN ***F****
Jun 6 15:04:21 194.217.188.53:7766 -> MY.NET.20.10:1359 NULL *1***** RESERVEDBITS
Jun 6 15:13:03 194.217.188.53:31514 -> MY.NET.20.10:31501 INVALIDACK **S*R*AU
Jun 6 15:13:23 194.217.188.53:1434 -> MY.NET.20.10:11677 INVALIDACK **S**PA*
Jun 6 15:13:25 194.217.188.53:27952 -> MY.NET.20.10:27960 SYNFIN **SF****
Jun 6 15:13:49 194.217.188.53:27990 -> MY.NET.20.10:1027 NULL *****
Jun 6 18:44:49 195.173.136.5:1298 -> MY.NET.20.10:1183 INVALIDACK **S*RPA*
Jun 6 18:44:49 195.173.136.5:27960 -> MY.NET.20.10:27960 NULL *****
Jun 6 18:44:50 195.173.136.5:53 -> MY.NET.20.10:53 FIN ***F****
Jun 6 22:01:32 195.173.128.94:3152 -> MY.NET.20.10:1290 INVALIDACK 21**RPAU RESERVEDBITS
```

All of these strange packets are generated by the infamous Demon-Net. It has been said that this network has misconfigured routers on it. The four FIN scans, looking for name servers inside the noise is particularly interesting... could this be a super stealth technique? Also interesting are the potential scans for Quake servers (destination port 27960)

Trace 15

```
Jun 6 11:09:51 24.28.193.160:6688 -> MY.NET.181.87:2034 NOACK 2*S*R*** RESERVEDBITS
Jun 6 11:12:35 24.28.193.160:6688 -> MY.NET.181.87:2034 NOACK 2*S*R*** RESERVEDBITS
Jun 6 15:53:34 24.226.115.234:1163 -> MY.NET.181.87:6688 NOACK ***FR***
Jun 6 16:11:27 209.237.79.241:53666 -> MY.NET.181.87:6688 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 16:22:52 146.102.92.25:85 -> MY.NET.181.87:2229 NOACK ****R**U
```

The first two traces show a Cable modem user sending strange packets to MY.NET.181.87. These packets do not conform to standard TCP/IP transactions, therefore they are considered anomalous. The destination port 2034 doesn't appear to be a Trojan port or known exploit port. The next three traces also have rather anomalous traffic going to

MY.NET.181.87. A possible conclusion could be a coordinated attack using packets that have either no ack flag set or an invalid ack flag set

Trace 16

```
Jun 6 16:57:29 213.224.0.48:1036 -> MY.NET.217.22:6346 NULL *****  
Jun 6 17:14:48 213.224.0.48:21081 -> MY.NET.217.22:6346 XMAS 21*F*P*U RESERVEDBITS
```

This combination of scans suggests that the tool was nmap. Null and Xmas scans are an attempt to bypass firewalls and intrusion detection systems

Trace 17

```
Jun 6 18:46:57 147.26.115.167:1069 -> MY.NET.217.102:6699 NOACK **S*RP**
```

The concern with this trace is that only one exists. Could this be a very low slow scan? The SYN, RESET, and PUSH flags could be a way to bypass an IDS or firewall and push data to the destination IP, like a covert channel using strange packets.

Trace 18

Jun 6 19:06:41 194.154.153.201:3354 -> MY.NET.97.148:1243 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3355 -> MY.NET.97.148:21554 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3356 -> MY.NET.97.148:1080 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3357 -> MY.NET.97.148:20034 SYN **S*****
Jun 6 19:06:40 194.154.153.201:3358 -> MY.NET.97.148:40421 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3359 -> MY.NET.97.148:31338 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3360 -> MY.NET.97.148:31785 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3362 -> MY.NET.97.148:9872 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3361 -> MY.NET.97.148:5400 SYN **S*****
Jun 6 19:06:40 194.154.153.201:3364 -> MY.NET.97.148:7307 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3351 -> MY.NET.97.148:12345 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3353 -> MY.NET.97.148:6670 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3352 -> MY.NET.97.148:31337 SYN **S*****
Jun 6 19:06:41 194.154.153.201:3366 -> MY.NET.97.148:61466 SYN **S*****
Jun 6 19:06:44 194.154.153.201:3363 -> MY.NET.97.148:20000 SYN **S*****
Jun 6 19:06:42 194.154.153.201:3359 -> MY.NET.97.148:31338 SYN **S*****
Jun 6 19:06:42 194.154.153.201:3356 -> MY.NET.97.148:1080 SYN **S*****
Jun 6 19:06:42 194.154.153.201:3357 -> MY.NET.97.148:20034 SYN **S*****
Jun 6 19:06:42 194.154.153.201:3354 -> MY.NET.97.148:1243 SYN **S*****
Jun 6 19:06:43 194.154.153.201:3366 -> MY.NET.97.148:61466 SYN **S*****
Jun 6 19:06:46 194.154.153.201:3358 -> MY.NET.97.148:40421 SYN **S*****
Jun 6 19:06:47 194.154.153.201:3361 -> MY.NET.97.148:5400 SYN **S*****
Jun 6 19:06:47 194.154.153.201:3351 -> MY.NET.97.148:12345 SYN **S*****
Jun 6 19:06:49 194.154.153.201:3361 -> MY.NET.97.148:5400 SYN **S*****
Jun 6 19:06:49 194.154.153.201:3351 -> MY.NET.97.148:12345 SYN **S*****

— A source IP maintained by SpiderNet, in Nicosia, Cypress, and attempts to probe MY.NET.97.148 for popular Trojans within a second.

Trace 19

Jun 6 20:08:57 130.149.41.70:3035 -> MY.NET.217.74:994 NOACK 21S*R*** RESERVEDBITS
Jun 6 20:10:11 130.149.41.70:3041 -> MY.NET.217.74:994 XMAS ***F*P*U
Jun 6 20:11:25 130.149.41.70:3041 -> MY.NET.217.74:994 XMAS ***F*P*U
Jun 6 20:13:23 130.149.41.70:3043 -> MY.NET.217.74:994 NULL *****
Jun 6 20:14:29 130.149.41.70:3043 -> MY.NET.217.74:994 NOACK 2**FRP*U RESERVEDBITS
Jun 6 20:15:17 130.149.41.70:3045 -> MY.NET.217.74:994 NOACK 21S*R*** RESERVEDBITS
Jun 6 20:15:27 130.149.41.70:3045 -> MY.NET.217.74:994 NOACK 21S*R*** RESERVEDBITS
Jun 6 20:15:31 130.149.41.70:202 -> MY.NET.217.74:3045 NOACK 21S*R*** RESERVEDBITS
Jun 6 20:21:35 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:21:44 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:22:16 130.149.41.70:0 -> MY.NET.217.74:3069 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:25:12 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:25:56 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:27:31 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:27:52 130.149.41.70:3069 -> MY.NET.217.74:994 INVALIDACK *1**R*AU RESERVEDBITS
Jun 6 20:39:49 130.149.41.70:3083 -> MY.NET.217.74:994 INVALIDACK ***FR*A*
Jun 6 20:43:14 130.149.41.70:246 -> MY.NET.217.74:3083 INVALIDACK ***FR*A*
Jun 6 20:49:07 130.149.41.70:3087 -> MY.NET.217.74:994 INVALIDACK *1SFR*AU RESERVEDBITS
Jun 6 20:50:43 130.149.41.70:3087 -> MY.NET.217.74:994 INVALIDACK *1SFR*AU RESERVEDBITS
Jun 6 20:54:17 130.149.41.70:3090 -> MY.NET.217.74:994 UNKNOWN 2****PAU RESERVEDBITS
Jun 6 20:54:46 130.149.41.70:3090 -> MY.NET.217.74:994 UNKNOWN 2****PAU RESERVEDBITS
Jun 6 20:55:00 130.149.41.70:3090 -> MY.NET.217.74:994 NULL *****
Jun 6 20:56:01 130.149.41.70:3090 -> MY.NET.217.74:994 INVALIDACK ***FRPAU
Jun 6 20:56:01 130.149.41.70:3090 -> MY.NET.217.74:994 UNKNOWN 2****PAU RESERVEDBITS
Jun 6 20:57:15 130.149.41.70:3090 -> MY.NET.217.74:994 NOACK ****RP**
Jun 6 20:58:39 130.149.41.70:3090 -> MY.NET.217.74:994 FULLXMAS 21SFRPAU RESERVEDBITS
Jun 6 21:12:30 130.149.41.70:3097 -> MY.NET.217.74:994 INVALIDACK 21S*RP*AU RESERVEDBITS
Jun 6 21:13:26 130.149.41.70:61 -> MY.NET.217.74:3097 INVALIDACK 21S*RP*AU RESERVEDBITS
Jun 6 21:14:51 130.149.41.70:3097 -> MY.NET.217.74:994 INVALIDACK 21S*RP*AU RESERVEDBITS
Jun 6 21:19:34 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS
Jun 6 21:20:10 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS
Jun 6 21:21:48 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS
Jun 6 21:24:48 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS
Jun 6 21:25:27 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS
Jun 6 21:25:34 130.149.41.70:3102 -> MY.NET.217.74:994 FULLXMAS 2*SFRPAU RESERVEDBITS

Jun 6 21:37:48 130.149.41.70:3107 -> MY.NET.217.74:994 INVALIDACK *1S*RPAU RESERVEDBITS
Jun 6 21:40:04 130.149.41.70:3107 -> MY.NET.217.74:994 INVALIDACK *1S*RPAU RESERVEDBITS
Jun 6 21:43:17 130.149.41.70:202 -> MY.NET.217.74:3123 NOACK ****R**U
Jun 6 21:43:22 130.149.41.70:3123 -> MY.NET.217.74:994 NOACK ****R**U
Jun 6 21:46:36 130.149.41.70:0 -> MY.NET.217.74:3123 NOACK ****R**U
Jun 6 21:46:55 130.149.41.70:3123 -> MY.NET.217.74:994 NOACK ****R**U
Jun 6 21:50:12 130.149.41.70:202 -> MY.NET.217.74:3135 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 21:50:29 130.149.41.70:3135 -> MY.NET.217.74:994 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 21:51:03 130.149.41.70:3135 -> MY.NET.217.74:994 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 21:53:19 130.149.41.70:173 -> MY.NET.217.74:3135 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 21:54:28 130.149.41.70:202 -> MY.NET.217.74:3135 UNKNOWN 21**R*A* RESERVEDBITS
Jun 6 21:57:52 130.149.41.70:3147 -> MY.NET.217.74:994 NOACK *1*FRP*U RESERVEDBITS
Jun 6 21:58:30 130.149.41.70:3147 -> MY.NET.217.74:994 NOACK 21SFRP** RESERVEDBITS
Jun 6 21:58:39 130.149.41.70:3147 -> MY.NET.217.74:994 INVALIDACK 21**R*AU RESERVEDBITS
Jun 6 21:59:47 130.149.41.70:3147 -> MY.NET.217.74:994 FIN ***F****
Jun 6 22:00:41 130.149.41.70:3147 -> MY.NET.217.74:994 NULL *****
Jun 6 22:01:32 195.173.128.94:3152 -> MY.NET.20.10:1290 INVALIDACK 21**RPAU RESERVEDBITS
Jun 6 22:02:01 130.149.41.70:3147 -> MY.NET.217.74:994 NOACK 21SFRP** RESERVEDBITS
Jun 6 22:03:17 130.149.41.70:3147 -> MY.NET.217.74:994 NOACK 21S**P** RESERVEDBITS
Jun 6 22:03:57 130.149.41.70:3147 -> MY.NET.217.74:994 NOACK 21SFRP** RESERVEDBITS
Jun 6 22:05:05 130.149.41.70:3150 -> MY.NET.217.74:994 INVALIDACK ***FR*A*
Jun 6 22:05:16 130.149.41.70:173 -> MY.NET.217.74:3150 INVALIDACK ***FR*A*
Jun 6 22:08:24 130.149.41.70:3150 -> MY.NET.217.74:994 NOACK 2*S*RP** RESERVEDBITS
Jun 6 22:12:08 130.149.41.70:3155 -> MY.NET.217.74:994 NOACK *1S**P** RESERVEDBITS
Jun 6 22:13:19 130.149.41.70:3155 -> MY.NET.217.74:994 NOACK *1S**P** RESERVEDBITS
Jun 6 22:14:07 130.149.41.70:202 -> MY.NET.217.74:3155 NOACK *1S**P** RESERVEDBITS
Jun 6 22:20:33 130.149.41.70:3158 -> MY.NET.217.74:994 VECNA 2*****U RESERVEDBITS
Jun 6 22:21:23 130.149.41.70:3158 -> MY.NET.217.74:994 NOACK *1*FRP*U RESERVEDBITS
Jun 6 22:21:25 130.149.41.70:3158 -> MY.NET.217.74:994 VECNA 2*****U RESERVEDBITS
Jun 6 22:22:43 130.149.41.70:3158 -> MY.NET.217.74:994 VECNA 2*****U RESERVEDBITS
Jun 6 22:23:07 130.149.41.70:3158 -> MY.NET.217.74:994 VECNA 2*****U RESERVEDBITS
Jun 6 22:23:07 130.149.41.70:61 -> MY.NET.217.74:3158 VECNA 2*****U RESERVEDBITS
Jun 6 22:26:04 130.149.41.70:3158 -> MY.NET.217.74:994 NOACK *1*FRP*U RESERVEDBITS
Jun 6 22:26:42 130.149.41.70:3158 -> MY.NET.217.74:994 NOACK *1*FRP*U RESERVEDBITS
Jun 6 22:27:22 130.149.41.70:3158 -> MY.NET.217.74:994 NOACK ***FRP**
Jun 6 22:27:40 130.149.41.70:3162 -> MY.NET.217.74:994 NULL *****
Jun 6 22:29:15 130.149.41.70:3162 -> MY.NET.217.74:994 NULL *****

Jun 6 22:33:47 130.149.41.70:3162 -> MY.NET.217.74:994 VECNA *****P*U

Many, many strange packets; many of the packets are targeted at MY.NET.217.74's port 994; could this be an attempt to create noise and then scan under the noise level... the FIN, NULL, XMAS and FULLXMAS packets could be an NMAP probe. One thing is evident.... A tool that is attempting to subvert MY.NET.217.74's TCP stack crafted these packets.

Trace 20

Jun 6 23:01:28 63.11.38.168:1461 -> MY.NET.97.177:21554 SYN **S*****
Jun 6 23:01:28 63.11.38.168:1463 -> MY.NET.97.177:456 SYN **S*****
Jun 6 23:01:30 63.11.38.168:1458 -> MY.NET.97.177:27374 SYN **S*****
Jun 6 23:01:30 63.11.38.168:1455 -> MY.NET.97.177:20034 SYN **S*****
Jun 6 23:01:29 63.11.38.168:1456 -> MY.NET.97.177:31377 SYN **S*****
Jun 6 23:01:29 63.11.38.168:1457 -> MY.NET.97.177:1243 SYN **S*****
Jun 6 23:01:31 63.11.38.168:1453 -> MY.NET.97.177:12345 SYN **S*****
Jun 6 23:01:31 63.11.38.168:1459 -> MY.NET.97.177:6670 SYN **S*****
Jun 6 23:01:31 63.11.38.168:1454 -> MY.NET.97.177:1080 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1511 -> MY.NET.97.177:1080 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1512 -> MY.NET.97.177:20034 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1516 -> MY.NET.97.177:6670 SYN **S*****
Jun 6 23:02:14 63.11.38.168:1515 -> MY.NET.97.177:27374 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1514 -> MY.NET.97.177:1243 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1513 -> MY.NET.97.177:31377 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1518 -> MY.NET.97.177:21554 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1519 -> MY.NET.97.177:9400 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1517 -> MY.NET.97.177:6671 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1520 -> MY.NET.97.177:456 SYN **S*****
Jun 6 23:02:15 63.11.38.168:1510 -> MY.NET.97.177:12345 SYN **S*****
Jun 6 23:05:18 63.11.38.168:1840 -> MY.NET.97.177:27374 SYN **S*****
Jun 6 23:05:17 63.11.38.168:1841 -> MY.NET.97.177:6670 SYN **S*****
Jun 6 23:05:19 63.11.38.168:1845 -> MY.NET.97.177:456 SYN **S*****
Jun 6 23:05:19 63.11.38.168:1842 -> MY.NET.97.177:6671 SYN **S*****

```
Jun 6 23:05:18 63.11.38.168:1838 -> MY.NET.97.177:31377 SYN **S*****  
Jun 6 23:05:18 63.11.38.168:1839 -> MY.NET.97.177:1243 SYN **S*****  
Jun 6 23:05:19 63.11.38.168:1843 -> MY.NET.97.177:21554 SYN **S*****  
Jun 6 23:05:19 63.11.38.168:1844 -> MY.NET.97.177:9400 SYN **S*****  
Jun 6 23:05:21 63.11.38.168:1835 -> MY.NET.97.177:12345 SYN **S*****
```

A source, which is from the IP address space maintained by UUNET, attempts to probe MY.NET.97.177 for popular Trojans within 4 seconds.

Upcoming Training

Click Here to
{Get CERTIFIED!}



SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
SANS vLive - SEC503: Intrusion Detection In-Depth	SEC503 - 201709,	Sep 11, 2017 - Oct 18, 2017	vLive
Baltimore Fall 2017 - SEC503: Intrusion Detection In-Depth	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Scottsdale SEC503	Scottsdale, AZ	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event
Community SANS Ottawa SEC503	Ottawa, ON	Oct 16, 2017 - Oct 21, 2017	Community SANS
SANS Berlin 2017	Berlin, Germany	Oct 23, 2017 - Oct 28, 2017	Live Event
San Diego Fall 2017 - SEC503: Intrusion Detection In-Depth	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
Community SANS Pensacola SEC503	Pensacola, FL	Nov 27, 2017 - Dec 02, 2017	Community SANS
SIEM & Tactical Analytics Summit & Training	Scottsdale, AZ	Nov 28, 2017 - Dec 05, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
SANS Dallas 2018	Dallas, TX	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced