Global Information Assurance Certification Paper

GIAC INTRUSION DETECTION PRACTICAL
TAMMY FLETCHER – NAVCIRT

Assignment 1 – Network Detects

Detect 1

SubSeven Trojan

> 00:43:21.657023 trolling.xxx.net.1886 > xxx.xxx.xxx.155.**27374**: S 202888701:202888701(0) win 8192 (DF) [tos 0x58]
> 00:43:21.664723 trolling.xxx.net.1887 > xxx.xxx.xxx.156.27374: S 202888703:202888703(0) win 8192 (DF) [tos 0x94]
> 00:43:21.673036 trolling.xxx.net.1888 > xxx.xxx.xxx.157.27374: S 202888705:202888705(0) win 8192 (DF) [tos 0xd0]
> 00:43:21.680740 trolling.xxx.net.1889 > xxx.xxx.xxx.158.27374: S 202888707:202888707(0) win 8192 (DF) [tos 0xc]
> 00:43:21.688972 trolling.xxx.net.1890 > xxx.xxx.xxx.159.27374: S 202888709:202888709(0) win 8192 (DF) [tos 0x48]
> 00:43:21.696793 trolling.xxx.net.1891 > xxx.xxx.xxx.160.27374: S 202888711:202888711(0) win 8192 (DF) [tos 0xe4]
> 00:43:21.704985 trolling.xxx.net.1892 > xxx.xxx.xxx.161.27374: S 202888713:202888713(0) win 8192 (DF) [tos 0x20]
> 00:43:21.712848 trolling.xxx.net.1893 > xxx.xxx.xxx.162.27374: S 202888715:202888715(0) win 8192 (DF) [tos 0x5c]

1. Source of Trace: Sanitized Incident report submitted to NAVCIRT by navy command
2. Detect was generated by: Shadow tcpdump
   Explanation of fields:

00:43:21.649118 trolling.xxx.net.1886 > xxx.xxx.xxx.154.27374: S 202888699:202888699
[timestamp] [source.port] [dest.port] [flags] [beg seq # : end seq #] [
(0) win 8192 (DF) [tos 0x98]
[bytes] [options] [type of service]

3. Probability the source address was spoofed: Source IP not spoofed
4. Description of Attack: Trolling for Trojans – specifically SubSeven 2.1
5. Attack Mechanism: SubSeven is a Trojan for the windows platform. It comes at least in two parts, a client and a server. The client is used by the hacker to connect to the victim' s machine. Once the server.exe is installed on the victim's machine the hacker has full access to the victim's machine.

   SubSeven download Files:
   server.exe- The real Trojan, which is installed on the victim's machine
   sub7.exe - The client used by the hacker to connect to his victim's machine
   EditServer.exe - configuration utility to set several configuration options on server.exe. Gives the hacker the opportunity to configure the port used by server.exe and to set a password for the server and to set some notification options, i.e. notify when his victim(s) are online. This notification can be done using ICQ, IRC, or e-mail.

   The popularity of scans for this Trojan is due to the fact that one victim can be commanded to scan for other victims. This has lead to numerous scans for port 27374 on the net.
   How a Trojan horse works:
   First stage of a Trojan Horse attack is to get the program on a user's machine. Typical techniques are:
   - post the program to newsgroups claiming to be some other program
   - spam mailing lists with the attached program
   - post program to websites
   - send via instant messenger programs and chat systems (ICQ, AIM, IRC, etc.)
   - forge e-mail from the ISP (like AOL) with a hoax message asking somebody to run a program (such as a software update).
   - copy to startup folder via "File and Print Sharing".

   The next stage of the attack is to scan the Internet looking for machines that might be compromised. The problem is that most of the techniques outlined above don't tell the cracker/hacker where their victim machine is. Therefore, the cracker/hacker must scan the Internet looking for the machines they might have compromised. This leads the condition where owners of firewalls (including personal firewalls) regularly see "probes" directed at their machines from crackers/hackers looking for these machines. However, if the machine hasn't been compromised, then these probes are not a problem. The probes cannot compromise the machine by themselves.

6. Correlation's:

Relevant CVE's / CVE candidates / references:

- Excerpts taken from http://www.robertgraham.com/pubs/firewall-seen.html#1.4
- http://advice.networkice.com/advice/Exploits/Ports/27374/default.htm
- CAN-1999-0572 .reg files are associated with the Windows NT registry editor, making the registry susceptible to Trojan Horse attacks.
- CAN-1999-0660 A hacker utility or Trojan Horse is installed on a system, e.g. NetBus, Back Orifice, Rootkit, etc.
- http://www.simovits.com/nyheter9902.html

7. Evidence of active targeting Not active targeting. Trolling.
8. Severity (Critical + Lethal) – (System + Net Countermeasures) = Severity
   (4+4)-(4+4)=0 Even if port 27374 is blocked at firewall, number of ways Trojan can infect systems such as email attachments.
9. Defensive recommendations: Perimeter defenses appear to be adequate. Ensure antivirus software is updated.
10. Multiple choice question:
    Besides 27374, what other ports are probable ports for the SubSeven Trojan?
    A. 1243
    B. 2140
    C. 31337
    D. 31335

Answer is A.

Detect 2

Scan for port 23 (telnet). Remote Login attempt.

```
> 18:07:19.221106 xxx.xxx.xxx.30.2247 > yyy.yyy.yyy.2.23: S 2733441058:2733441058(0) win 32120 (DF)
> 18:07:19.222908 xxx.xxx.xxx.30.2248 > yyy.yyy.yyy.3.23: S 2728141059:2728141059(0) win 32120 (DF)
> 18:07:19.224875 xxx.xxx.xxx.30.2249 > yyy.yyy.yyy.4.23: S 2730166079:2730166079(0) win 32120 (DF)
> 18:07:19.226186 xxx.xxx.xxx.30.2250 > yyy.yyy.yyy.5.23: S 2733215216:2733215216(0) win 32120 (DF)
> 18:07:19.228029 xxx.xxx.xxx.30.2251 > yyy.yyy.yyy.6.23: S 2728163802:2728163802(0) win 32120 (DF)
> 18:07:19.228683 xxx.xxx.xxx.30.2246 > yyy.yyy.yyy.1.23: S 2743676169:2743676169(0) win 32120 (DF)
> 18:07:19.229462 xxx.xxx.xxx.30.2252 > yyy.yyy.yyy.7.23: S 2736192797:2736192797(0) win 32120 (DF)
> 18:07:19.230936 xxx.xxx.xxx.30.2253 > yyy.yyy.yyy.8.23: S 2733438891:2733438891(0) win 32120 (DF)
> 18:07:19.233067 xxx.xxx.xxx.30.2254 > yyy.yyy.yyy.9.23: S 2731388855:2731388855(0) win 32120 (DF)
> 18:07:19.235196 xxx.xxx.xxx.30.2255 > yyy.yyy.yyy.10.23: S 2738040285:2738040285(0) win 32120 (DF)
> 18:07:19.236342 xxx.xxx.xxx.30.2256 > yyy.yyy.yyy.11.23: S 2740450235:2740450235(0) win 32120 (DF)
> 18:07:19.237531 xxx.xxx.xxx.30.2257 > yyy.yyy.yyy.12.23: S 2728537895:2728537895(0) win 32120 (DF)
> 18:07:19.239005 xxx.xxx.xxx.30.2258 > yyy.yyy.yyy.13.23: S 2728925684:2728925684(0) win 32120 (DF)
> 18:07:19.240602 xxx.xxx.xxx.30.2259 > yyy.yyy.yyy.14.23: S 2731358289:2731358289(0) win 32120 (DF)
```

1. Source of Trace: Sanitized Incident report submitted to NAVCIRT by navy command.
2. Detect was generated by: Shadow tcpdump
   Explanation of fields:

```
18:07:19.240602  xxx.xxx.xxx.30.2259 > yyy.yyy.yyy.14.23:  S        2731358289:2731358289    (0)      win 32120  (DF)
[timestamp]      [source.port]          [dest.port]          [flag]   [beg seq # : end seq #]  [bytes]  [options]
```

3. Probability the source address was spoofed: Source IP probably not spoofed.
Probing for active service.
4. Description of Attack: Telnet probe
5. Attack Mechanism: Automated scan of network – port 23. Source IP sending SYN to all system on Network in attempt to establish telnet connection. Unsuccessful – no reply from any system on the network.
6. Correlation's:
Relevant CVE's and / or CVE candidates / references:
- http://www.sans.org/newlook/resources/IDFAQ/telnet_rlogin.htm

- CAN-1999-0285 Denial of service in telnet from the Windows NT Resource Kit, by opening then immediately closing a connection.
- CVE-1999-0073 Telnet allows a remote client to specify environment variables including LD_LIBRARY_PATH, allowing an attacker to bypass the normal system libraries and gain root access.
- CVE-1999-0087 Denial of service in AIX telnet can freeze a system and prevent users from accessing the server.
- CVE-1999-0192 Buffer overflow in telnet daemon telnet routing allows remote attackers to gain root access via the TERMCAP environmental variable.

7. Evidence of active targeting: Not active targeting. Scan of entire network. Reconnaissance effort.
8. Severity: (Critical + Lethal) – (System + Net Countermeasures) = Severity
   (4+5)-(3+5)=1 possibly lethal if successfully establishes telnet connection and login.
9. Defensive recommendations: Ensure logon restrictions are in place and restricted to internal network. Use SSH vice telnet for remote logon external to the network. Perimeter defenses appear to be adequate as there were no responses to attack.
10. Multiple choice question:
    Telnet service is assigned what port?
    A. 21
    B. 20
    C. 23
    D. 25

Answer is C - port 23.

Detect 3

SYN/FIN scan

> 10:42:19.206344 xxx.0.0.99.**9704** > x.y.z.1.**9704**: **SF 850048815:850048815**(0) win 1028
> 10:42:19.235506 xxx.0.0.99.9704 > x.y.z.2.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.264996 xxx.0.0.99.9704 > x.y.z.3.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.265651 xxx.0.0.99.9704 > x.y.z.4.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.295305 xxx.0.0.99.9704 > x.y.z.5.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.326926 xxx.0.0.99.9704 > x.y.z.6.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.327621 xxx.0.0.99.9704 > x.y.z.7.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.355268 xxx.0.0.99.9704 > x.y.z.8.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.386231 xxx.0.0.99.9704 > x.y.z.9.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.387378 xxx.0.0.99.9704 > x.y.z.10.9704: SF 850048815:850048815(0) win 1028
> 10:42:19.417033 xxx.0.0.99.9704 > x.y.z.11.9704: SF 850048815:850048815(0) win 1028

1. Source of Trace: Sanitized Incident report submitted to NAVCIRT by navy command.
2. Detect was generated by: Shadow tcpdump
   Explanation of fields:

10:42:19.417033	xxx.0.0.99.9704 > x.y.z.11.9704:	SF	850048815:850048815	(0)	win 1028
[timestamp]	**[sourc.port]**	**[dest.port]**	**[flag]**	**[beg seq # : end seq #]**	**[bytes]**	**[options]**

3. Probability the source address was spoofed: Probably not spoofed. SYN/FIN scan used for intelligence gathering.
4. Description of Attack: SYN / FIN scan
5. Attack Mechanism: Automated attack. Foreign source. Packet appears to be crafted– SF flags set, source port number does not increment and sequence numbers do not change. Port 9704 is a port added to the inetd to cause an overflow for rpc.statd . SYN and FIN flags set simultaneously is an anomalous condition (not logical). FINS maybe allowed thru a filtering device even if the SYN's are not, which improves the probability of a response. Since FIN tears down the connection some logging systems may not report the connection attempt. Goal of a SYN / FIN packet is to penetrate the firewall.
   This scan has been identified as part of an Rpc.statd exploit. First, a list of target addresses are scanned looking for a response on port 23 or 25 which is an attempt to identify the OS. A list of redhat linux machines is created from the responses and a batch file is executed that sends a command to insert a root shell into /etc/inetd.conf on port 9704.
6. Correlation's:

Relevant CVE's and / or CVE candidates / references:

- CVE-2000-0666 - rpc.statd in the nfs-utils package in various Linux distributions does not properly cleanse untrusted format strings, which allows remote attackers to gain root privileges.
- http://www.sans.org/082200.htm
- http://lists.insecure.org/incidents/2000/Aug/0170.html
- http://archives.neohapsis.com/archives/incidents/current/0001.html
- http://archives.neohapsis.com/archives/incidents/current/0067.html
- http://www.securityfocus.com/bid/1480
- http://www.cert.org/advisories/CA-2000-17.html
- Pg 96-99 Network Intrusion Detection (Stephen Northcutt)

7. Evidence of active targeting: Based on description of port 9704 exploit this appears to be active targeting. Scan across entire network looking for active rootshell on port 9704.
8. Severity: (Critical + Lethal) – (System + Net Countermeasures) = Severity
   (2+2)-(4+4)=2
9. Defensive recommendations: Perimeter defenses appear to be adequate as there were no responses to attack.
10. Multiple choice question:

What indicates that this might be a crafted packet?

A. SF flags set
B. Non-incrementing sequence numbers

C. Source port stays same
D. All of the above

Answer is D.

Detect 4

PINGMAP

> 06:54:39.922903 xxx.xxx.galactica.it > dest.xxx.32.255: icmp: echo request
> 06:54:39.925721 xxx.xxx.galactica.it > dest.xxx.32.0: icmp: echo request
> 06:54:39.935969 xxx.xxx.galactica.it > dest.xxx.33.255: icmp: echo request
> 06:54:39.938390 xxx.xxx.galactica.it > dest.xxx.33.0: icmp: echo request
> 06:54:39.949074 xxx.xxx.galactica.it > dest.xxx.34.255: icmp: echo request
> 06:54:39.950209 xxx.xxx.galactica.it > dest.xxx.34.0: icmp: echo request
> 06:54:39.958465 xxx.xxx.galactica.it > dest.xxx.35.0: icmp: echo request
> 06:54:39.969926 xxx.xxx.galactica.it > dest.xxx.35.255: icmp: echo request
> 06:54:39.977405 xxx.xxx.galactica.it > dest.xxx.36.0: icmp: echo request
> 06:54:39.978072 xxx.xxx.galactica.it > dest.xxx.36.255: icmp: echo request
> 06:54:39.988819 xxx.xxx.galactica.it > dest.xxx.37.0: icmp: echo request
> 06:54:39.989583 xxx.xxx.galactica.it > dest.xxx.37.255: icmp: echo request
> 06:54:40.013297 xxx.xxx.galactica.it > dest.xxx.38.255: icmp: echo request
> 06:54:40.015733 xxx.xxx.galactica.it > dest.xxx.38.0: icmp: echo request
> 06:54:40.023784 xxx.xxx.galactica.it > dest.xxx.39.0: icmp: echo request
> 06:54:40.025094 xxx.xxx.galactica.it > dest.xxx.39.255: icmp: echo request
> 06:54:40.033479 xxx.xxx.galactica.it > dest.xxx.40.255: icmp: echo request
> 06:54:40.035280 xxx.xxx.galactica.it > dest.xxx.40.0: icmp: echo request

1. Source of Trace: Sanitized Incident report submitted to NAVCIRT by navy command.
2. Detect was generated by: Shadow tcpdump
   Explanation of fields:

06:54:40.035280 xxx.xxx.galactica.it > dest.xxx.40.0: icmp: echo request
[timestamp] [source] [destination] [icmp:icmp message]

3. Probability the source address was spoofed: Used as initial Reconnaissance effort. Source IP probably not spoofed.
4. Description of Attack: Pingmap
5. Attack Mechanism: Network mapping is an intelligence gathering phase – attempts to discover the IP's of live hosts in the target network. ICMP supports broadcast traffic so scanner in above detect is attempting to send an ICMP echo request to the broadcast addresses .0 and .255 of the target networks. If ICMP is allowed, all active hosts within that broadcast subnet may reply. Broadcast 0 is an archaic broadcast but Unix and other systems will often answer to it. Windows systems will answer the 255 broadcast. A reply will possibly allow the attacker to identify type of system.
6. Correlation's:
   Relevant CVE's and / or CVE candidates / references:
   - CVE-1999-0513 ICMP messages to broadcast addresses are allowed, allowing for a Smurf attack that can cause a denial of service.
   - also reference - Pg 125 Network Intrusion Detection (Stephen Northcutt)
   - http://www.sans.org/newlook/resources/IDFAQ/traffic.htm
7. Evidence of active targeting: Automated attack of dest.xxx network but no specific target.
8. Severity: (Critical + Lethal) – (System + Net Countermeasures) = Severity
   (2+1)-(3+3)= 3
9. Defensive recommendations: Block ICMP echo requests at firewall or filtering router. Disallow activity to broadcast addresses. Perimeter defenses appear to be good – no response to attack.
10. Multiple choice question:
    Which system will answer to the 255 broadcast?
    A. Unix
    B. Linux
    C. Windows
    D. Printers

    Answer is C.

Assignment 2 – Evaluate an Attack

- Location attack was aquired from:

Attack generated on local network. NMAP downloaded from **http://www.insecure.org/nmap/**
Nmap is covered under the GNU General Public License (GPL) and can be downloaded free of charge.

- Describe attack and how it works:

NMAP is a powerful information gathering tool . It is designed to scan large networks to determine which hosts are up and what services they are offering, readily available tool to the hacker community.

Generated NMAP scan of port 80 with option set to discover Operating System on local network.

- nmap -v -P0 -sS -p 80 victim.of.love.com –O

options: -v verbose

-PO Do not try and ping hosts at all before scanning them. This allows the scanning of networks that don't allow ICMP echo requests (or responses) through their firewall.

-sS TCP SYN scan: This technique is often referred to as "half-open" scanning, because you don't open a full TCP connection. You send a SYN packet, as if you are going to open a real connection and you wait for a response. A SYN|ACK indicates the port is listening. A RST is indicative of a non-listener. If a SYN|ACK is received, a RST is immediately sent to tear down the connection

The primary advantage to this scanning technique is that fewer sites will log it. Stealth scanning.

-p <port ranges> This option specifies what ports you want to Scan. The default is to scan all ports between 1 and 1024 as well as any ports listed in the services file which comes with nmap.

–O This option activates remote host identification via TCP/IP fingerprinting. In other words, it uses a bunch of techniques to detect subtleties in the underlying operating system network stack of the computers you are scanning. It uses this information to create a 'fingerprint' which it compares with its database of known OS fingerprints (the nmap-os-fingerprints file) to decide what type of system you are scanning. Fingerprinting the TCP stack includes such techniques as FIN probing to see what kind of response the target has, BOGUS flag probing to see the remote host's reaction to undefined flags sent with a SYN packet, TCP Initial Sequence Number (ISN) sampling to find patterns of ISN numbers, as well as other methods of determining the remote operating system. A definitive article on stack fingerprinting, written by Fyodor, the author of Nmap, can be found at http://www.insecure.org/nmap/nmap-fingerprinting-article.html.

- Annotated network trace of attack in action:

Below is screen shot from source.

```
> nmap -v -P0 -sS -p 80 victim.of.love.com –O
Starting nmap V. 2.3BETA6 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Initiating SYN half-open stealth scan against (victim.of.love.com)
Adding TCP port 80 (state Open).
The SYN scan took 0 seconds to scan 1 ports.
For OSScan assuming that port 80 is open and port 39782 is closed and neither are firewalled
Interesting ports on (victim.of.love.com):
Port State Protocol Service
80 open tcp http

TCP Sequence Prediction: Class=random positive increments
                         Difficulty=5693905 (Good luck!)

Sequence numbers: F007A8FA F0959FB0 EFC2ECFD F0ACA711 F0A6AEBD F06B0169
Remote operating system guess: Linux 2.1.122 - 2.2.12
OS Fingerprint:
TSeq(Class=RI%gcd=1%SI=56E1D1)
T1(Resp=Y%DF=Y%W=7F53%ACK=S++%Flags=AS%Ops=MENNTNW)
T2(Resp=N)
T3(Resp=Y%DF=Y%W=7F53%ACK=S++%Flags=AS%Ops=MENNTNW)
```

T4(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T5(Resp=Y%DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T7(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
PU(Resp=Y%DF=N%TOS=C0%IPLEN=164%RIPTL=148%RID=E%RIPCK=E%UCK=E%ULEN=134%DAT=E)

Nmap run completed -- 1 IP address (1 host up) scanned in 1 second

TCPDUMP output on destination:

16:36:23.924073 < ill.get.you.com.50644 > victim.of.love.com.www: S 2911718787:2911718787(0) win 2048
16:36:23.924186 > victim.of.love.com.www > ill.get.you.com.50644: S 4033517201:4033517201(0) ack 2911718788 win 32696 <mss 536> (DF)
16:36:23.924682 < ill.get.you.com.50644 > victim.of.love.com.www: R 2911718788:2911718788(0) win 0
16:36:23.926167 < ill.get.you.com.50651 > victim.of.love.com.www: S [ECN-Echo] 910504423:910504423(0) win 2048 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.926222 > victim.of.love.com.www > ill.get.you.com.50651: S 4022502439:4022502439(0) ack 910504424 win 32595 <mss 265,nop,nop,timestamp 1191089 1061109567,nop,wscale 0> (DF)
16:36:23.926299 < ill.get.you.com.50652 > victim.of.love.com.www: . 910504423:910504423(0) win 2048 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.926501 < ill.get.you.com.50653 > victim.of.love.com.www: SFP 910504423:910504423(0) win 2048 urg 0 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.926546 > victim.of.love.com.www > ill.get.you.com.50653: S 4021884165:4021884165(0) ack 910504424 win 32595 <mss 265,nop,nop,timestamp 1191089 1061109567,nop,wscale 0> (DF)
16:36:23.926659 < ill.get.you.com.50654 > victim.of.love.com.www: . 910504423:910504423(0) ack 0 win 2048 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.926717 > victim.of.love.com.www > ill.get.you.com.50654: R 0:0(0) win 0
16:36:23.926802 < ill.get.you.com.50651 > victim.of.love.com.www: R 910504424:910504424(0) win 0
16:36:23.927104 < ill.get.you.com.50653 > victim.of.love.com.www: R 910504424:910504424(0) win 0
16:36:23.927285 < ill.get.you.com.50655 > victim.of.love.com.39782: S 910504423:910504423(0) win 2048 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.927352 > victim.of.love.com.39782 > ill.get.you.com.50655: R 0:0(0) ack 910504424 win 0
16:36:23.927416 < ill.get.you.com.50656 > victim.of.love.com.39782: . 910504423:910504423(0) ack 0 win 2048 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.927471 > victim.of.love.com.39782 > ill.get.you.com.50656: R 0:0(0) win 0
16:36:23.927770 < ill.get.you.com.50657 > victim.of.love.com.39782: FP 910504423:910504423(0) win 2048 urg 0 <wscale 10,nop,mss 265,timestamp 1061109567 0,eol>
16:36:23.927820 > victim.of.love.com.39782 > ill.get.you.com.50657: R 0:0(0) ack 910504423 win 0
16:36:23.928440 < ill.get.you.com.50644 > victim.of.love.com.39782: udp 300
16:36:23.928732 > victim.of.love.com > ill.get.you.com: icmp: victim.of.love.com udp port 39782 unreachable [tos 0xc0]
16:36:24.710196 < ill.get.you.com.50645 > victim.of.love.com.www: S 910504424:910504424(0) win 2048
16:36:24.710247 > victim.of.love.com.www > ill.get.you.com.50645: S 4027033850:4027033850(0) ack 910504425 win 32696 <mss 536> (DF)
16:36:24.710636 < ill.get.you.com.50645 > victim.of.love.com.www: R 910504425:910504425(0) win 0
16:36:24.730182 < ill.get.you.com.50646 > victim.of.love.com.www: S 910504425:910504425(0) win 2048
16:36:24.730228 > victim.of.love.com.www > ill.get.you.com.50646: S 4036337584:4036337584(0) ack 910504426 win 32696 <mss 536> (DF)
16:36:24.730609 < ill.get.you.com.50646 > victim.of.love.com.www: R 910504426:910504426(0) win 0
16:36:24.750181 < ill.get.you.com.50647 > victim.of.love.com.www: S 910504426:910504426(0) win 2048
16:36:24.750229 > victim.of.love.com.www > ill.get.you.com.50647: S 4022529277:4022529277(0) ack 910504427 win 32696 <mss 536> (DF)
16:36:24.750609 < ill.get.you.com.50647 > victim.of.love.com.www: R 910504427:910504427(0) win 0
16:36:24.770186 < ill.get.you.com.50648 > victim.of.love.com.www: S 910504427:910504427(0) win 2048
16:36:24.770233 > victim.of.love.com.www > ill.get.you.com.50648: S 4037846801:4037846801(0) ack 910504428 win 32696 <mss 536> (DF)
16:36:24.770619 < ill.get.you.com.50648 > victim.of.love.com.www: R 910504428:910504428(0) win 0
16:36:24.790254 < ill.get.you.com.50649 > victim.of.love.com.www: S 910504428:910504428(0) win 2048
16:36:24.790302 > victim.of.love.com.www > ill.get.you.com.50649: S 4037455549:4037455549(0) ack 910504429 win 32696 <mss 536> (DF)
16:36:24.790685 < ill.get.you.com.50649 > victim.of.love.com.www: R 910504429:910504429(0) win 0
16:36:24.810185 < ill.get.you.com.50650 > victim.of.love.com.www: S 910504429:910504429(0) win 2048
16:36:24.810233 > victim.of.love.com.www > ill.get.you.com.50650: S 4033544553:4033544553(0) ack 910504430 win 32696 <mss 536> (DF)
16:36:24.810633 < ill.get.you.com.50650 > victim.of.love.com.www: R 910504430:910504430(0) win 0
16:36:28.920209 < arp who-has victim.of.love.com tell ill.get.you.com
16:36:28.920271 > arp reply victim.of.love.com (0:60:97:3c:af:1) is-at 0:60:97:3c:af:1 (0:0:c0:58:8f:f4)

Probing for port 80 – Source sends a SYN packet as if going to open real connection. Victim responds with a SYN/ACK to indicate that it is listening on port 80. Source immediately sends R (reset) to tear down connection.

Probing for Operating System :

Source sends combination of bogus flags, FIN probes, TCP ISN Sampling, TCP Initial window size, ACK value, various TCP options for example and responses from victim create a "fingerprint" which Is compared to database of know OS fingerprints and a best guess is made. In this case NMAP Remote operating system guess was Linux 2.1.122 - 2.2.12. Actual operating system of victim is *Red Hat Linux release 6.2 (Zoot)*.

Assignment 3 – "Analyze this" scenario

Analysis of Snort detects for approximately one month. Standard rule base.
Snort detects obtained from http://www.sans.org/PH2000/snort/index.htm
Snort Alert logs start 29 June 00 and end 06 Aug 00.
Snort Scan logs start 30 June 00 and end 10 Aug 00.
Data is incomplete due to power failures and / or system problems – i.e. disk full.

Utilized SnortSnarf v10094001.1 to analyze data.
(available at www.silicondefense.com/snortsnarf/main.html)

Generated following with Snortsnarf – see assignment 4 for details.

362199 alerts processed.

Signature	# Alerts	# Sources	# Destinations
FTP-bad-login	1	1	1
Telnet daemon-active	1	1	1
PING-ICMP Source Quench	1	1	1
Back Orifice	1	1	1
Possible wu-ftpd exploit	2	1	2
Queso fingerprint	3	3	3
Happy 99 Virus	4	4	4
wu-ftpd exploit	5	3	4
large ICMP Packet	5	5	1
TELNET - Login Incorrect	7	3	6
External RPC call	8	2	1
Tiny Fragments-Possible Hostile activity	9	3	3
Napster Client Data	12	8	7
SUNRPC highport access!	18	3	3
Null scan!	30	20	19
NMAP TCP ping!	45	6	5
Napster 7777 Data	170	14	13
GIAC 000218 VA-CIRT port 35555	182	28	9
GIAC 000218 VA-CIRT port 34555	196	25	9
SMB Name Wildcard	229	5	4
Napster 8888 Data	323	8	8
SNMP public access	1147	28	1
MISC - Large UDP Packet	1170	1	1
WinGate 1080 Attempt	2042	353	305
Attempted Sun RPC high port access	2241	10	8
WinGate 8080 Attempt	3222	89	16
Watchlist 000222 NET-NCFC	4711	40	12
PING-ICMP Time Exceeded	6689	299	117
PING-ICMP Destination Unreachable	12313	133	144
Watchlist 000220 IL-ISDNNET-990517	13962	19	17
SYN-FIN scan!	19844	11	19801

TABLE 1

Reviewed results for Snortsnarf and found following items of interest:

Most active sources:			
# of alerts	**IP address**	**type of alarm/exploit**	**whois**
1	MY.NET.99.51	ACTIVE TELNET DAEMON	
1	209.245.5.158	ICMP SOURCE QUENCH	
1	202.159.46.234	BACKORRIFICE	INDONET, INDONESIA
2	151.164.223.206	WU-FTPD	SOUTHWESTERN BELL,TX
1	24.3.29.155	QUESO FINGERPRINT	@ HOME , MD
1	210.84179.196	QUESO FINGERPRINT	OZEMAIL2-AU
1	192.203.80.142	QUESO FINGERPRINT	RUSSIAN ACADEMY OF SCI
1	203.251.136.2	HAPPY 99 VIRUS	KOREA TELECOM
1	200.223.11.7	HAPPY 99 VIRUS	RNP BRAZIL
1	206.67.51.242	HAPPY 99 VIRUS	MEDIA 3 TECH
1	208.130.42.17	HAPPY 99 VIRUS	LOGON AMERICA
6	63.236.34.174	TINY FRAGMENTS	QUOKA SPORTS
14	205.188.3.205	SUNRPC HIGH PORT ACCESS	AOL
3	210.121.242.164	NULL SCAN	KOREA TELECOM
5	149.225.111.69	NULL SCAN	AUNET, DE
23	205.128.11.157	NMAP TCP PING	HEADHUNTER NET
90	208.184.216.183	NAPSTER 7777 DATA	ABOVENET
14	207.217.120.29	GAIC VA-CIRT PORT 35555	EARTHLINK
63	152.163.224.100	GIAC VA-CIRT PORT 34555	AOL
219	MY.NET.101.160	SMB NAME WILDCARD	
205	208.184.216.189	NAPSTER 8888 DATA	ABOVENET
131	MY.NET.97.237	SNMP PUBLIC ACCESS	
159	MY.NET.97.80	SNMP PUBLIC ACCESS	
208	MY.NET.97.186	SNMP PUBLIC ACCESS	
1170	211.40.176.214	large UDP packet	BORANET KOREA
155	168.120.16.250	WINGATE 1080	ASSUMPTION UNIVERSITY, TH
104	208.240.218.220	WINGATE 1080	PROF. COMPUTER SERVICES
2166	205.188.153.111	SUNRPC	AOL, VIRGINIA
1145	128.231.171.123	WINGATE 8080	National Inst of Health (1 DEST)
275	24.3.26.53	WINGATE	@ HOME MD, CATV (1 Dest)
222	216.0.124.26	WINGATE	DIGEX INC, MD
19818	202.0.178.98	SYN/FIN	China Motion Telcom Holdings Ltd.
4923	24.23.96.119 (PA)	dest unreachable	@Home Network
2346	24.4.52.197 (TX)	dest unreachable	@Home Network
801	MY.NET.14.2	ICMP time exceeded	MY.NET (112 destinations)
Most active destinations:			
# of alerts	**IP address**	**type of alarm/exploit**	**whois**
1	24.25.111.117	TIME WARNER ROADRUNNER MN	
1	MY.NET.70.121	ICMP SOURCE QUENCH	
1	MY.NET.100.100	BACKORRIFICE	
1	MY.NET.99.16	WU-FTPD	
1	MY.NET.144.59	WU-FTPD	
1	MY.NET.60.8	QUESO	
1	MY.NET.6.44	QUESO	
1	MY.NET.99.23	QUESO	
1	MY.NET.110.150	HAPPY 99	
1	MY.NET.253.42	HAPPY 99	
1	MY.NET.6.47	HAPPY 99	
1	MY.NET.6.34	HAPPY 99	

6	MY.NET.1.8	TINY FRAGMENTS
14	MY.NET.98.145	SUNRPC HIGH PORT ACCESS
5	MY.NET.60.14	NULL SCAN
4	MY.NET.100.236	NULL SCAN
34	MY.NET.1.8	NMAP
90	MY.NET.97.204	NAPSTER 7777
74	MY.NET.253.24	GIAC 000218 35555
115	MY.NET.253.24	GIAC 000218 34555
219	MY.NET.101.192	SMB NAME WILDCARD
249	MY.NET.201.2	NAPSTER 8888
1147	MY.NET.101.192	SNMP PUBLIC ACCESS
1170	MY.NET.98.179	LARGE UDP PACKET
150	MY.NET.60.16	WINGATE 1080
241	MY.NET.60.8	WINGATE 1080
285	MY.NET.60.11	WINGATE 1080
2166	MY.NET.217.126	SUNRPC
2854	MY.NET.253.105	WINGATE (51 SOURCES)
11305	MY.NET.70.121	dest unreachable
271	MY.NET.140.9	dest unreachable
5830	MY.NET.140.9	ICMP Time exceeded

TABLE 2

********NOTEWORTHY EVENTS **************

1. NAPSTER (port 6699)

08/01-01:52:09.897907 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.38.141:2792 -> MY.NET.217.38:6699

Jul 27 12:45:02 24.112.193.183:6699 -> MY.NET.182.71:2334 NOACK 2*S*R*** RESERVEDBITS
Jul 27 13:32:23 24.166.184.108:2116 -> MY.NET.98.107:6699 INVALIDACK ****R*AU

08/05-18:30:07.112277 [**] Napster 8888 Data [**] MY.NET.201.2:1463 -> 208.184.216.191:8888

Aug 4 12:35:38 193.150.235.135:52547 -> MY.NET.20.10:8888 SYN **S*****
Aug 10 17:34:03 64.244.202.66:62949 -> MY.NET.179.86:8888 SYN **S*****

08/05-18:34:08.606042 [**] Napster 7777 Data [**] MY.NET.97.229:49153 -> 208.184.216.178:7777
08/05-18:34:09.147293 [**] Napster 7777 Data [**] 208.184.216.178:7777 -> MY.NET.97.229:49153

Jul 9 21:26:06 165.138.228.4:7777 -> MY.NET.97.68:2077 UDP
Jul 9 21:26:06 165.138.228.4:7777 -> MY.NET.97.68:2079 UDP

Possibly being exploited – foreign source (Israel) attempting to connect to Napster client.

Users share MP3 files from Napster.com. Destination port 6699 is common.
Napster uses TCP for client to server communication. Typically the servers run on ports 8888 and 7777.
The gnapster port (version 1.3.8 and earlier), and the knapster port (version 0.9 and earlier) contain a vulnerability which allows remote napster users to view any file on the local system which is accessible to the user running gnapster/knapster.
Relevent CVE candidates:

CAN-2000-0281 ** Buffer overflow in the Napster client beta 5 allows remote attackers to cause a denial of service via a long message.

CAN-2000-0412 ** The gnapster and knapster clients for Napster do not properly restrict access only to MP3 files, which allows remote attackers to read arbitrary files from the client by specifying the full pathname for the file.

2. Destination port 34555 – 182 alerts (28 sources to 9 destinations)
Destination port 35555 – 196 alerts (25 sources to 9 destinations)

07/14-17:24:07.681717 [**] GIAC 000218 VA-CIRT port 34555 [**] 165.251.8.74:25-> MY.NET.253.24:34555
07/14-17:24:07.820670 [**] GIAC 000218 VA-CIRT port 34555 [**] 165.251.8.74:25-> MY.NET.253.24:34555

07/14-18:22:19.700023 [**] GIAC 000218 VA-CIRT port 35555 [**] 207.69.200.243:113 -> MY.NET.253.43:35555
07/17-19:05:44.909909 [**] GIAC 000218 VA-CIRT port 35555 [**] 132.239.1.48:113 -> MY.NET.100.230:35555

DDOS – distributed denial of service utilized by TRINOO – risk rating medium – virus type Trojan.
Daemon agent that runs on windows platform.
Relevent CVE candidates:
CAN-2000-0138 ** A system has a distributed denial of service (DDOS) attack master, agent, or zombie installed, such as (1) Trinoo, (2) Tribe Flood Network (TFN), (3) Tribe Flood Network 2000 (TFN2K), (4) stacheldraht, (5) mstream, or (6) shaft.

3. Back Orifice
Foreign source - Indonesia.

07/12-17:16:32.897041 [**] Back Orifice [**] 202.159.46.234:31338 -> MY.NET.100.130:31337

BackOrifice is a program that allows hackers to access and even control someone else's PC, over the Internet. It was released in August of 1998 by a group of hackers calling themselves The Cult of the Dead Cow (they call it a "remote administration tool). BackOrifice can only affect a machine on which it's been deliberately installed, and it works only on computers running Windows 95 or 98. Once you detect BackOrifice, you can neutralize it fairly quickly. To find out whether or not BackOrifice is installed on your machine, you can search your hard drive for a file called "windll.dll," which BackOrifice creates whenever it runs

Relevent CVE candidates:
CAN-1999-0660
** CANDIDATE (under review) ** A hacker utility or Trojan Horse is installed on a system, e.g. NetBus, Back Orifice, Rootkit, etc.
CAN-2000-0562
** CANDIDATE (under review) ** BlackIce Defender 2.1 and earlier, and BlackIce Pro 2.0.23 and earlier, do not properly block Back Orifice traffic when the security setting is Nervous or lower.

4. Wingate 1080 SOCKS
155 Wingate scans from foreign source.

07/14-00:03:20.138859 [**] WinGate 1080 Attempt [**] 168.120.16.250:55067 -> MY.NET.97.135:1080
07/14-00:04:04.529242 [**] WinGate 1080 Attempt [**] 203.155.129.248:4387 -> MY.NET.97.135:1080

Most scans for port 1080 are actually looking for WinGate, a popular firewall/proxy for Windows
Relevent CVE's:
CVE-1999-0290 The WinGate telnet proxy allows remote attackers to cause a denial of service via a large number of connections to localhost.
CVE-1999-0291 The WinGate proxy is installed without a password, which allows remote attackers to redirect connections without authentication.
CVE-1999-0441 Remote attackers can perform a denial of service in WinGate machines using a buffer overflow in the Winsock Redirector Service.
CVE-1999-0494 Denial of service in WinGate proxy through a buffer overflow in POP3.

5. NMAP ping

07/28-23:32:23.408944 [**] NMAP TCP ping! [**] 216.127.150.136:57882 -> MY.NET.253.114:1
08/04-08:01:02.191197 [**] NMAP TCP ping! [**] 195.25.86.2:80 -> MY.NET.179.77:80
08/04-10:49:10.811041 [**] NMAP TCP ping! [**] 205.128.11.157:80 -> MY.NET.1.8:53
08/04-10:49:10.811088 [**] NMAP TCP ping! [**] 205.128.11.157:53 -> MY.NET.1.8:53
08/04-11:18:28.348261 [**] NMAP TCP ping! [**] 205.128.11.157:80 -> MY.NET.1.8:53
08/04-11:18:28.348302 [**] NMAP TCP ping! [**] 205.128.11.157:53 -> MY.NET.1.8:53

07/12-12:46:34.921774 [**] Probable NMAP fingerprint attempt [**] 24.200.160.45:1548 -> MY.NET.70.241:8899

NMAP is a powerful information gathering tool. It is designed to scan large networks to determine which hosts are up and what services they are offering, readily available tool to the hacker community.

6. Happy 99 Virus

07/19-04:28:40.867369 [**] Happy 99 Virus [**] 203.251.136.2:4985 -> MY.NET.253.42:25
07/26-07:50:56.700210 [**] Happy 99 Virus [**] 208.130.42.17:40221 -> MY.NET.6.34:25
08/05-11:22:48.017066 [**] Happy 99 Virus [**] 206.67.51.242:4889 -> MY.NET.6.47:25
07/11-19:28:57.652242 [**] Happy 99 Virus [**] 200.223.11.7:4836 -> MY.NET.110.150:25

When executed, the infected program opens a window entitled "Happy New Year 1999 !!" and shows a firework display to disguise its installation. Then it spams itself to the same newsgroups or same e-mail addresses where the user was posting or mailing to. It maps SKA.EXE to memory and converts it to uuencoded format and mails an additional e-mail
or newsgroup post with the same header information as the original message but containing no text but just an attachment called Happy99.exe.
see http://www.europe.F-Secure.com/v-descs/ska.htm

7. SunRPC

07/19-14:26:12.632395 [**] Attempted Sun RPC high port access [**] 24.4.129.16:407 -> MY.NET.115.91:32771
07/19-14:26:12.632451 [**] Attempted Sun RPC high port access [**] 24.4.129.16:1419 -> MY.NET.115.91:32771

Connection attempts are being made to port 32771. Under Solaris, the Rpcbind service listens on port 32771 in addition to the standard port 111. It is very likely that the attackers are attempting to connect to this service in order to find out what RPC services are being offered. There are several known buffer overflow vulnerabilities with RPC services that can be exploited to grant root access.

8. NULL SCAN

07/14-12:28:25.838842 [**] Null scan! [**] 24.232.51.137:1152 -> MY.NET.110.57:6688
07/14-12:28:29.384871 [**] Null scan! [**] 24.232.51.137:1152 -> MY.NET.110.57:6688

07/28-10:46:17.044983 [**] spp_portscan: PORTSCAN DETECTED from 213.6.123.12 (STEALTH) [**]
07/28-10:30:11.805484 [**] Null scan! [**] 213.6.123.12:6699 -> MY.NET.182.94:3419
07/28-10:46:19.361501 [**] spp_portscan: portscan status from 213.6.123.12: 1 connections across 1 hosts: TCP(1), UDP(0)

Null Scan. A TCP frame has been seen with a sequence number of zero and all control bits are set to zero.
This frame should never be seen in normal TCP operation. A hacker may be scanning your system by sending these specially formatted frames to see what services are available. Sometimes this is done in preparation for a future attack, or sometimes it is done to see if your system might have a service which is susceptible to attack
See http://advice.networkice.com/Advice/Intrusions/2000309/default.htm

9. SNMP public access

07/14-08:13:19.325170 [**] SNMP public access [**] MY.NET.97.237:1042 -> MY.NET.101.192:161
07/14-08:13:26.293120 [**] SNMP public access [**] MY.NET.97.237:1044 -> MY.NET.101.192:161

Internal network traffic indicates that default community string “public” has not been renamed.
This can allow internal users to map the network.
Relevant CVE’s / candidates:
- CVE-1999-0294 All records in a WINS database can be deleted through SNMP for a denial of service.
- CVE-1999-0472 The SNMP default community name "public" is not properly removed in NetApps C630 Netcache, even if the administrator tries to disable it.
- CAN-1999-0516 ** An SNMP community name is guessable.
- **CAN-1999-0517 ** An SNMP community name is the default (e.g. public), null, or missing.**

10. SMB Name Wildcard

07/14-16:28:41.808896 [**] SMB Name Wildcard [**] MY.NET.101.160:137 -> MY.NET.101.192:137
07/14-16:29:54.218468 [**] SMB Name Wildcard [**] MY.NET.101.160:137 -> MY.NET.101.192:137

The SMB logon request contains the size of data which follows. When the size of data which is specified in the request does not correspond to the size of data which is actually present, corruption occurs. Malicious users can launch denial of service attacks against Microsoft Windows NT systems.

11. Watchlists

07/14-16:09:40.239312 [**] Watchlist 000222 NET-NCFC [**] 159.226.49.23:4552 -> MY.NET.145.9:25
CHINA

07/17-11:51:24.229184 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.4.238:1072 -> MY.NET.53.28:4110
ISRAEL

Although watchlists were in place for activity from China / Israel still seeing ample amount of suspicious activity from these sources. Might possible recommend IP blockage.

12. SYN-FYN scan

19844 alerts (11 sources - 202.0.178.98 foreign – China triggered 19818 of those alerts)

07/29-13:06:49.354956 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.3:21
07/29-13:06:49.369729 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.4:21
07/29-13:06:49.400032 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.5:21
07/29-15:30:46.393217 [**] SYN-FIN scan! [**] 212.177.241.139:80 -> MY.NET.1.5:80
07/29-17:51:09.959388 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.3:21
07/29-17:51:09.973574 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.4:21
07/29-17:51:09.993383 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.5:21
07/29-17:52:43.151531 [**] SYN-FIN scan! [**] 212.177.241.139:109 -> MY.NET.1.3:109

07/29-13:06:49.354956 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.3:21
07/29-13:06:49.369729 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.4:21
07/29-13:06:49.400032 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.5:21
07/29-15:30:46.393217 [**] SYN-FIN scan! [**] 212.177.241.139:80 -> MY.NET.1.5:80
07/29-17:51:09.959388 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.3:21
07/29-17:51:09.973574 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.4:21
07/29-17:51:09.993383 [**] SYN-FIN scan! [**] 208.50.27.150:21 -> MY.NET.1.5:21
07/29-17:52:43.151531 [**] SYN-FIN scan! [**] 212.177.241.139:109 -> MY.NET.1.3:109

06/28-06:52:55.149077 [**] SYN-FIN scan! [**] **202.0.178.98:53** -> MY.NET.2.88:53
06/28-06:52:55.263743 [**] SYN-FIN scan! [**] 202.0.178.98:53 -> MY.NET.2.93:53
06/28-06:52:55.268354 [**] SYN-FIN scan! [**] 202.0.178.98:53 -> MY.NET.2.95:53
06/28-06:52:55.319495 [**] SYN-FIN scan! [**] 202.0.178.98:53 -> MY.NET.2.98:53

SYN FIN scan – POP II buffer overflow vulnerability exists if source / dest ports both 109 or both 53. SYN and FIN flags set simultaneously is an anomalous condition (not logical). FINS maybe allowed thru a filtering device even if the SYN's are not, which improves the probability of a response. Since FIN tears down the connection some logging systems may not report the connection attempt. Goal of a SYN / FIN packet is to penetrate the firewall.

13. Large UDP packet 1170 alerts from 1 source to 1 destination

08/05-18:30:03.777730 [**] IDS247 - MISC - Large UDP Packet [**] 211.40.176.214:29536 -> MY.NET.98.179:6970
08/05-18:30:03.835886 [**] IDS247 - MISC - Large UDP Packet [**] 211.40.176.214:29536 -> MY.NET.98.179:6970

An abnormally large UDP packet was detected – this may indicate a denial of service attack. May indicate possible covet channels of communication such as backdoors or control traffic for DDOS.

Following Systems possibly compromised:

MY.NET.1.3

07/19-09:49:16.702569 [**] Queso fingerprint [**] 212.171.169.46:24122 -> MY.NET.1.3:21

Jul 29 13:06:49 208.50.27.150:21 -> MY.NET.1.3:21 SYNFIN **SF****

07/14-13:48:58.394814 [**] spp_portscan: portscan status from MY.NET.1.3: 10 connections across 2 hosts: TCP(0), UDP(10) [**]

Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41909 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41910 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41911 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41912 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41913 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41914 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41916 UDP

MY.NET.1.3 performed multitude of scans against it's own network after being scanned and fingerprinted.

MY.NET.1.8

06/27-07:39:33.390475 [**] NMAP TCP ping! [**] 209.218.228.46:80 -> MY.NET.1.8:53
06/27-07:39:33.390629 [**] NMAP TCP ping! [**] 209.218.228.46:53 -> MY.NET.1.8:53
07/08-07:21:32.145547 [**] Attempted Sun RPC high port access [**] 64.27.29.2:2385 -> MY.NET.1.8:32771
07/08-07:33:06.203162 [**] Attempted Sun RPC high port access [**] 207.230.26.34:1295 -> MY.NET.1.8:32771
07/08-20:02:37.444826 [**] NMAP TCP ping! [**] 209.218.228.46:80 -> MY.NET.1.8:53

MY.NET.1.8 was victim of reconnaissance thru NMAP and then SunRPC high port access was achieved.
42 alerts going to MY.NET.1.8

MY.NET.99.51

07/26-02:46:25.820700 [**] WinGate 1080 Attempt [**] 207.114.4.46:3875 -> MY.NET.99.51:1080
07/28-05:44:51.442479 [**] WinGate 1080 Attempt [**] 207.114.4.46:1272 -> MY.NET.99.51:1080
08/05-19:03:45.522918 [**] IDS08 - TELNET - daemon-active [**] MY.NET.99.51:23-> 24.25.111.117:1029
06/29-04:40:46.546586 [**] WinGate 1080 Attempt [**] 207.114.4.46:3816 -> MY.NET.99.51:1080
06/30-05:54:34.091505 [**] WinGate 1080 Attempt [**] 207.114.4.46:4360 -> MY.NET.99.51:1080
MY.NET.99.51 recieved numerous WinGate proxy scans. Telnet daemon indicates successful telnet connection has been established from outside local network. Telnet very insecure protocol and should be replaced with SSH.

MY.NET.253.114

07/28-23:32:11.772069 [**] WinGate 1080 Attempt [**] 216.127.150.136:1856 -> MY.NET.253.114:1080
07/28-23:32:23.368753 [**] Null scan! [**] 216.127.150.136:57878 -> MY.NET.253.114:22
07/28-23:32:23.408944 [**] NMAP TCP ping! [**] 216.127.150.136:57882 -> MY.NET.253.114:1
08/05-13:37:20.335822 [**] SUNRPC highport access! [**] 209.138.185.157:4067 -> MY.NET.253.114:32771
08/05-13:37:20.337188 [**] SUNRPC highport access! [**] 209.138.185.157:4067 -> MY.NET.253.114:32771

SUMMARY:

MY.NET network was the target of numerous distributed denial of service attacks, reconnaissance efforts, SYN floods, and possible malicious activity with the virus and Trojan signatures. Appears that there was a major DOS attack on 05 Aug 00. The presence of WatchList indicates past history of suspicious activity from Israel and China but still see evidence of suspicious activity thru firewall. Site needs to ensure that antivirus software is current; ACL's are accurate and configured to silently drop unauthorized activity. ICMP disabled when not needed. Napster deinstalled and not authorized for usage. Disable port 1080 unless actually a proxy server. Ensure all security software patches are installed. This is just a snapshot of potential vulnerabilities based on Snort files provided. Due to lack of chronological logs for all days some activity could not be correlated that might have significant value.

Assignment 4 – Analyze Process

Utilized SnortSnarf v10094001.1 to analyze data.
(available at www.silicondefense.com/snortsnarf/main.html - created by (Jim Hoagland and Stuart Staniford))
Downloaded SnortSnarf-100400.1.tar.gz
Gunziped file and then extracted the tar file.

Merged alert logs into single file SnortMergA.txt. Merged scan logs into single SnortMergS.txt.

Replaced MY.NET with 255.254 and ran perl script. Had to run snortsnarf.pl in the /SnortSnarf-100400-1/include directory because of permissions. SnortSnarf script takes the output files from Snort and converts them into webpages. Opened Netscape and viewed the index.htm which is displayed in Table 1 of assignment 3. Reviewed SnortSnarf output and created Table 2 which is a list of the most active sources / type of alert and whois information. Thoroughly reviewed the data provided in the SnortSnarf tables / links. Reviewed the two WatchLists and paid particular attention to unusual activity from these sites. By grepping the SnortMergA.txt and SnortMergS.txt for specific IP's or ports was able to establish activity correlation / trends I used to identify the boxes suspected of being compromised. For example:

- more SnortMergA.txt | grep MY.NET.1.3

07/19-09:49:16.702569 [**] Queso fingerprint [**] 212.171.169.46:24122 -> MY.NET.1.3:21

Jul 29 13:06:49 208.50.27.150:21 -> MY.NET.1.3:21 SYNFIN **SF****

07/14-13:48:58.394814 [**] spp_portscan: portscan status from MY.NET.1.3: 10 connections across 2 hosts: TCP(0), UDP(10) [**]

Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41909 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41910 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41911 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41912 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41913 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41914 UDP
Aug 5 10:32:29 MY.NET.1.3:53 -> MY.NET.101.89:41916 UDP

MY.NET.1.3 performed multitude of scans against it's own network after being scanned and fingerprinted

Used the following references to assist in analysis:

http://www.simovits.com/nyheter9902.html ports used by Trojans
http://www.cve.mitre.org
http://www.securityfocus.com
http://bugtraq.com
http://www.robertgraham.com/pubs/firewall-seen.html
http://www.sans.org/newlook/resources/
Network Intrusion Detection, An Analyst's Handbook by Stephen Northcutt

Last Updated: September 20th, 2017

Upcoming Training

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
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
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 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 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 Munich December 2017	Munich, Germany	Dec 04, 2017 - Dec 09, 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 London February 2018	London, United Kingdom	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Dallas 2018	Dallas, TX	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	Tysons, VA	Mar 17, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, Australia	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS 2018	Orlando, FL	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MD	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced