Global Information Assurance Certification Paper

Mark Evans

GIAC Intrusion Analyst: Practical Assignment

Part One: Five Attacks Analysed

1 Network Analysis One

Source of Trace:

External network, outside the Firewall. Snort IDS deployed. Multiple firewalls, defence in depth techniques used.

Detect was generated by:

Snort Intrusion Detection system. Snort was running the Whitehats ArachNIDS rule set. The rule set has been tuned to suit the local environment. Supporting Tcpdump traffic is also provided.

The Snort IDS alerts on an incoming UDP traceroute to the primary DNS server (w.x.y.z) on the DNS UDP port 53. The source address is 194.72.87.200

> whois -h whois.ripe.net 194.72.87.200
>
> inetnum: 194.72.86.0 – 194.72.87.255
> netname: BT-CUST-43
> descry: Springboard Internet Services Limited

A reverse DNS lookup of the specific address provides the following DNS name: horlic.delphi.co.uk.

The format of the IDS alerts below is: Date & Time, Snort-Sensor-Name, Whitehats IDS Unique Number and Description, Source IP Address and Port, Destination Port Address.

Mar 23 15:15:17 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> **w.x.y.z:53**
Mar 23 17:31:16 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:**1025** -> w.x.y.z:53
Mar 23 17:42:35 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53

Mar 24 18:04:41 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:**1025** -> w.x.y.z:53

Mar 27 **08:45:12** ids1.target.co.nz snort: IDS115/Traceroute UDP:

194.72.87.200:**1025** -> w.x.y.z:53
Mar 27 **08:45:12** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
[25 alerts from the same second removed]
Mar 27 08:45:12 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 08:45:12 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 **08:45:54** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:**1025** -> w.x.y.z:53
Mar 27 **08:45:54** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
[19 alerts from the same second removed]
Mar 27 08:45:54 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 08:45:54 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53

Mar 27 **09:16:26** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:**1025** -> w.x.y.z:53
Mar 27 09:16:26 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 **10:30:57** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 **10:30:57** ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
[38 alerts from the same second removed]
Mar 27 10:30:57 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 16:22:14 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 16:22:14 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53
Mar 27 16:22:14 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53

Mar 28 01:19:08 ids1.target.co.nz snort: IDS115/Traceroute UDP:
194.72.87.200:1025 -> w.x.y.z:53

Examining the first packet, Tcpdump data shows a reverse lookup (PTR) for the address a.b.18.29. The packet has a time-to-live (TTL) of 1.

The format of the Tcpdump data below is: Time, Source Address and Port, Destination Address and Port, Payload (in this case a DNS PTR lookup) and payload length, Time to Live and the IP ID number.

15:15:17.898688 < **194.72.87.200**.1025 > **w.x.y.z.domain**: 57572 PTR? **d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 37888)

whois -h whois.arin.net a.b.c.d

ABCD Limited (NET-ABCD) Private Bag Auckland, NZ
Netname: ABCD Netblock: a.b.0.0 – a.b.255.255
Domain System inverse mapping provided by:
DNS.ABCD.CO.NZ a.b.18.29
dns.target.CO.NZ w.x.y.z

The address a.b.c.d resolves to dns.abcd.co.nz. This system is the primary DNS Server for ABCD Ltd. The targeted DNS server is listed as the secondary DNS server for ABCD's netblock.

Further Tcpdump data shows consistent interest in this netblock, with TTLs set to 1:

17:31:15.910763 < 194.72.87.200.1025 > w.x.y.z.domain: 47397 **PTR? d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 47163)

17:42:34.918636 < 194.72.87.200.1025 > w.x.y.z.domain: 14378 **PTR? d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 480)

18:04:41.106037 < 194.72.87.200.1025 > w.x.y.z.domain: 62373 **PTR? d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 47027)

21:06:28.124572 < 194.72.87.200.1025 > w.x.y.z.domain: 31982 **PTR? d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 20543)

21:55:56.319564 < 194.72.87.200.1025 > w.x.y.z.domain: 49931 **PTR? d.c.b.a.in-addr.arpa**. (44) [**ttl 1**] (id 28203)

The source address appears to be trying to reverse resolve addresses inside ABCD's netblock.

Possibility the source address was spoofed:

This is UDP traffic, no TCP three-way handshake is performed. The source address could be spoofed. However the DNS queries request answers. Further information outlined below suggests that the source address is not spoofed.

Description of Attack:

The TTL is suspicious; a UDP request that results in a TTL of 1 is conceivable. However 103 occurrences with the same TTL in a five-day period is rather unlikely. We do not see any further traffic from this source address.

The client port is suspicious. The port is locked at 1025 for over five days. These ports would normally be expected to change (ports above 1024 are known as ephemeral).

The frequency of the incoming traffic is suspicious also; over 40 incoming packets in a single second then nothing for many hours. It should also be noted that almost all

the traffic recorded was during working hours, New Zealand time. As such, this traffic (if not spoofed) originated overnight in the UK.

103 stimulus packets do not represent an effective denial of service unless there is an asymmetric response to these packets or a vulnerability that they exploit.

Attack Mechanism:

This is stimulus traffic. The service targeted is the DNS server. The DNS server in use is bind. Bind has been subject to a large number of patches to address vulnerabilities and exploits.

It is possible to generate a Denial of Service attack by spoofing a DNS UDP query. As the request packet size is less than the typical response packet size for this request, this would be an asymmetric attack. However the low frequency negates this.

Correlations:

None

Evidence of active targeting:

This was active targeting. Continued probing of the same single system on the same source and destination ports from a single system. DNS requests to a DNS server.

Severity:

Target Criticality = 5.
This is the primary DNS server

Attack Lethality = 1.
This is fairly low noise fingerprinting

System Countermeasures = 1
The DNS server is required to respond to the request

Network Countermeasures = 4
Dual Firewalls and IDS in use

Attack Severity = 1 (5+1) – (1 +4)

Defence Recommendations:

This is reconnaissance. The version of BIND in use is current, but must be maintained that way.

Question

Perimeter defences should always block malformed TCP packets on the external interface, except:

A) When the source is a business partner
B) When the destination address is in the DMZ
C) When the source is a root DNS server

D) There are no exceptions

Answer: D. Packets that are out of spec should be blocked. There is no justification for genuine traffic to be malformed. Checksums will prevent undetected damage in transit. Blocking on the external interface of the perimeter defence ensures that no other systems see this malformed traffic.

2 Network Analysis Two

Source of Trace:

Home network, with ADSL modem (performing address translation) and an external Check Point Firewall-1 4.1 SP3 system (also performing address translation).

An Internal Firewall (running a different firewall product) protects internal systems.

External and Internal IDS are deployed.

Detect was generated by:

Checkpoint Firewall-1 log file analysis.

The firewall logs on dropped incoming packets. The Firewall is connected to an ADSL modem. Both ADSL and cable modem networks are common places for vulnerability scans.

The Firewall log is shown below (note Check Point's log viewer interprets port 5362 as pcANYWHERE-stat). PC Anywhere is a popular piece of remote control software, (see http://www.symantec.com for further information).

The source addresses are 210.54.a.b and 210.54.x.y. The external firewall address is shown as 192.168.1.2.

The format of the Firewall-1 log data below is: Time, Action taken, Firewall name, Ethernet card ID, Protocol Type, Source Address & Port, Destination Address & Port. The dates are shown in separate records.

Date: Mar 26, 2001
12:07:47 drop Firewall_Ext_NAT >eth1 proto udp
src **210.54.a.b** dst **192.168.1.2** service **pcANYWHERE-stat**
12:14:50 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.a.b dst 192.168.1.2 service pcANYWHERE-stat
14:22:03 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.a.b dst 192.168.1.2 service pcANYWHERE-stat

Date: Mar 27, 2001
21:01:27 drop Firewall_Ext_NAT >eth1 proto udp
src **210.54.x.y** dst 192.168.1.2 service pcANYWHERE-stat
21:10:43 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:13:19 **accept** Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:20:21 accept Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:23:08 accept Firewall_Ext_NAT >eth1 proto udp

src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:37:12 accept Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:42:47 accept Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
21:56:03 **drop** Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
22:04:41 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
22:13:49 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat

Date: Mar 28,2001
6:23:02 drop Firewall_Ext_NAT >eth1 proto udp
src **210.54.x.y** dst 192.168.1.2 service pcANYWHERE-stat
6:24:58 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
6:53:43 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
7:08:29 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat
7:36:11 drop Firewall_Ext_NAT >eth1 proto udp
src 210.54.x.y dst 192.168.1.2 service pcANYWHERE-stat

There was no further traffic from these source addresses (or against this port) in the following seven days.

During the period of time when the Firewall **accept**ed the traffic (from Mar 27 21:13:19 until some time before 21:56:03 the same day), the Snort sensor issued the following alerts:

The format of the IDS alerts below is: Date & Time, Snort-Sensor-Name[process ID], IDS Unique Number and Description, from Source IP Address

Mar 27 21:13:19 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> **210.54.x.y**
Mar 27 21:20:21 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y
Mar 27 21:23:08 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y
Mar 27 21:25:29 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y
Mar 27 21:27:00 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y
Mar 27 21:37:12 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y
Mar 27 21:42:47 192.168.1.2 snort[3068]: ICMP Destination Unreachable (Undefined Code!): 192.168.1.2 -> 210.54.x.y

Note the Snort log timestamps map directly onto the Firewall accepts. This is to be expected. Snort has triggered on the Firewall sending destination unreachable messages back to the source. This is because the Firewall was temporarily address translating all incoming traffic to an internal host. This host did not have a route back to the source at the time (it was during a period of maintenance).

Possibility the source address was spoofed:

Unlikely. The source addresses resolve to names associated with the ADSL provider in use.

```
b.a.54.210.in-addr.arpa     name = b-a-126-173.adsl.xtra.co.nz
y.x.54.210.in-addr.arpa     name = y-x-255-149.adsl.xtra.co.nz
```

The ADSL connection used provides dynamically allocated addresses with a limited lease time. The address changes when the ADSL modem is power cycled. It is possible that two different ADSL systems attempted the same attack. However, since this attack signature does not appear in the Firewall logs for the last three months, it is unlikely that these are different systems.

ADSL and cable modem space is likely to include poorly configured systems, perhaps running PC Anywhere unsecured. This is often referred to as a 'target rich environment'.

In order for a PC Anywhere attack (as opposed to reconnaissance for active PC Anywhere systems) to be useful the source address cannot not be spoofed.

Description of Attack:

The external firewall was repeatedly scanned for UDP port 5632 associated with PC Anywhere.

An administrative error with the Firewall configuration allowed this port (and a few others) into the Firewall for less than one hour. The packets were still logged.

The external IDS detected the side effect (ICMP Destination Unreachable packets) of the traffic. The internal IDS did not alert on the PC Anywhere traffic (because it did not make it past the internal firewall).

Attack Mechanism:

These are stimuli packets. The service targeted is PC Anywhere. The security associated with this product has historically not been well respected Many systems do not have any access security on the PC Anywhere software. It is often configured to run as an administrative account.

Success in reaching a PC Anywhere equipped system is likely to lead to immediate host compromise.

This looks like an attempted exploit. However the default behaviour of the Java based

PC Anywhere client (on start up) is to scan the Class C network (from it's perspective) for PC Anywhere servers, so it can show them in the browser window. This may be just a badly configured client with an ADSL modem attached to it.

Correlations:

None. However the attack is plausible and there is no easy way of gaining correlation on a relatively small ADSL network. The ADSL provider might provide further information if pressed.

Evidence of active targeting:

If this was an attack from an NZ based ADSL system against an NZ based ADSL system, then it is targeted (but poorly).

If it is poor client configuration then the scan is a side effect of the software, i.e. it is not really targeted.

Severity:

The severity is assessed here on the basis of it being a genuine attack.

Target Criticality = 5.
This is the external firewall.

Attack Lethality = 1.
A PC Anywhere attack will not affect a Checkpoint Firewall-1 system on Linux

System Countermeasures = 5
The Firewall does not (and cannot) run PC Anywhere. It was not listening on port 5362.

Network Countermeasures = 5/1
The firewall is configured to drop this traffic and log it. Snort is set to alert on this traffic. Assessment = 5
The external firewall was misconfigured for a period of 43 minutes. During this time the traffic was allowed onto the Firewall (port 5362 was still not listening on the firewall). Assessment = 1
The internal Firewall would have dropped this traffic at all times. The internal IDS would have triggered (as the attack profile is in the IDS signature file) if the traffic had reached the internal network.

Attack Severity = -4 (5+1) – (5 +5) or **0** (5+1) – (1+5) for a 43 minute period.

Defence Recommendations:

This attack demonstrates the value of defence in depth (i.e. multiple firewalls) and of careful assessment of firewall rule set changes.

The maintenance procedures and change control for the external firewall should be reviewed. The source address cannot realistically be blocked (as it is dynamic).

The internal firewall should explicitly drop port 5362, as it protects Microsoft Windows based systems that are potentially capable of (but were not) running PC Anywhere.

Question

It is advisable to run PC Anywhere on Firewalls and related security infrastructure because:

A) Remote control software allows you to manage your firewall over the Internet.
B) It is not advisable to run PC Anywhere on Firewalls.
C) Check Point's Firewall-1 recognises PC Anywhere traffic and so blocks it automatically.
D) Remote Users of PC Anywhere do not have any privileges over the PC Anywhere server by default.

Answer: B. Whilst PC Anywhere and similar systems might ease the administrative burden, the security implications outweigh the potential benefits. Firewalls should be single task devices.

3 Network Analysis Three

Source of Trace:

Home network, with ADSL modem (performing address translation) and an external Check Point Firewall-1 4.1 SP3 system (providing address translation).

An Internal Firewall (running a different firewall product) protects internal systems.

External and Internal IDS are deployed

Detect was generated by:

Snort Intrusion Detection system. Snort was running the standard rule set. The rule set has been tuned to suit the local environment. Supporting Tcpdump traffic is also provided.

The format of the IDS alerts below is: Date & Time, Snort-Sensor-Name [process ID], IDS Unique Number and Description, from Source IP Address. "Stealth" indicates that the scan was spread out over a significant amount of time and that the ports were scanned in a pseudo-random order. This is done to evade IDS systems and firewalls that automatically block on port scans[1].

The source addresses is 151.20.197.75. External firewall address shows as 192.168.1.2.

Mar 17 20:34:00 192.168.1.2 snort[604]: spp_portscan: PORTSCAN DETECTED from **151.20.197.75** (STEALTH)
Mar 18 02:01:05 192.168.1.2 snort[604]: spp_portscan: portscan status from 151.20.197.75: 1 connections across 1 hosts: TCP(1), UDP(0) STEALTH
Mar 18 06:55:26 192.168.1.2 snort[604]: spp_portscan: End of portscan from 151.20.197.75: TOTAL time(0s) hosts(1) TCP(1) UDP(0) STEALTH

```
whois -h whois.arin.net 151.20.197.75

inetnum: 151.20.0.0 – 151.20.255.255
netname: LIBERO-INFOSTRADA
descr:   Free Internet Dial-up Services
```

The DNS lookup of the address provides ppp-75-197-20-151.libero.it.

Snort also recorded one of the above packets in its out-of-spec (oos) log. This occurs when a packet has an illegal combination of flags set. In this case both SYN and FIN were set. The format of the oos log is:

Date & Time, Source Address & Port, Destination Address & Port, Nature of illegality of packet

[1] Watchguard's Firebox (and others) can do this automatically.

Mar 17 20:34:00 **151.20.197.75**:111 -> 192.168.1.2:111 SYNFIN ******SF

There was further corroborating data in the Checkpoint Firewall-1 log. The format below is : Time, Action, Firewall name, Ethernet card ID, Protocol Type, Source Address & Port, Destination Address & Port.

20:34:00 drop Firewall_Ext_NAT >eth1 proto tcp
src **151.20.197.75** dst 192.168.1.2 service sunrpc

Possibility the source address was spoofed:

Unlikely. This is TCP traffic against portmapper on 111. This is unlikely to be spoofed as it is looking for a vulnerable system with port 111 open, and would need to complete the three-way handshake to exploit portmapper vulnerabilities.

Description of Attack:

Basic scans against the portmapper port. The portmapper service will identify what services are running (eg rpc.statd) and what ports they are listening on. This allows an attacker to quickly identify what exploits should be targeted at what services and where. If portmapper does provide this information then this constitutes a low noise piece of reconnaissance.

Attack Mechanism:

This is stimulus traffic. The SYN-FIN flags are set to try to avoid IDS detection or to fool the Firewall. Compromise of port 111 (portmapper) can lead to subsequent host compromise. This is an attempted exploit.

Correlations:

Detection by both Firewall and IDS sensor, otherwise none. SANS has highlighted increased scanning activity on port 111 throughout March (http://www.sans.org/y2k/archive-mar01.htm).

Evidence of active targeting:

Part of a scan of the address space for vulnerable systems with active portmapper services.

Severity:

Target Criticality = 5.
This is the external firewall.

Attack Lethality = 4
Portmapper vulnerabilities can help provide system access.

System Countermeasures = 5
The Firewall does not run the portmapper service.

Network Countermeasures = 5
The firewall drops port 111. This is a specific standalone rule with logging enabled.

Snort alerted on the attack pattern.

Attack Severity = -1 (5+4) – (5 +5)

Defence Recommendations:

Defences are fine, attack was blocked at the firewall (and logged). Snort detected the stealthy scan and the out-of-spec packets.

Question

Which of the above traffic captures proves the traffic was not a normal, benign TCP session

A) The source address resolves to a DNS name
B) The source address does not resolve to a DNS name
C) The SYN & FIN flags are not both permitted to be set on a given packet
D) Port 111 was blocked at the firewall

Answer: C. The TCP specification in the various RFCs preclude SYN and FIN being set on the same packet.

4 Network Analysis Four

Source of Trace:

External network, outside the Firewall. Snort IDS deployed. Multiple firewalls, defence in depth techniques used.

Detect was generated by:

Snort Intrusion Detection system. Snort was running the Whitehats rule set. The rule set has been tuned to suit the local environment.

The format of the IDS alerts below is: Date & Time, Snort-Sensor-Name[process ID], IDS Unique Number and Description, Source IP Address, Destination IP Address.

Snort alerts – Part One:

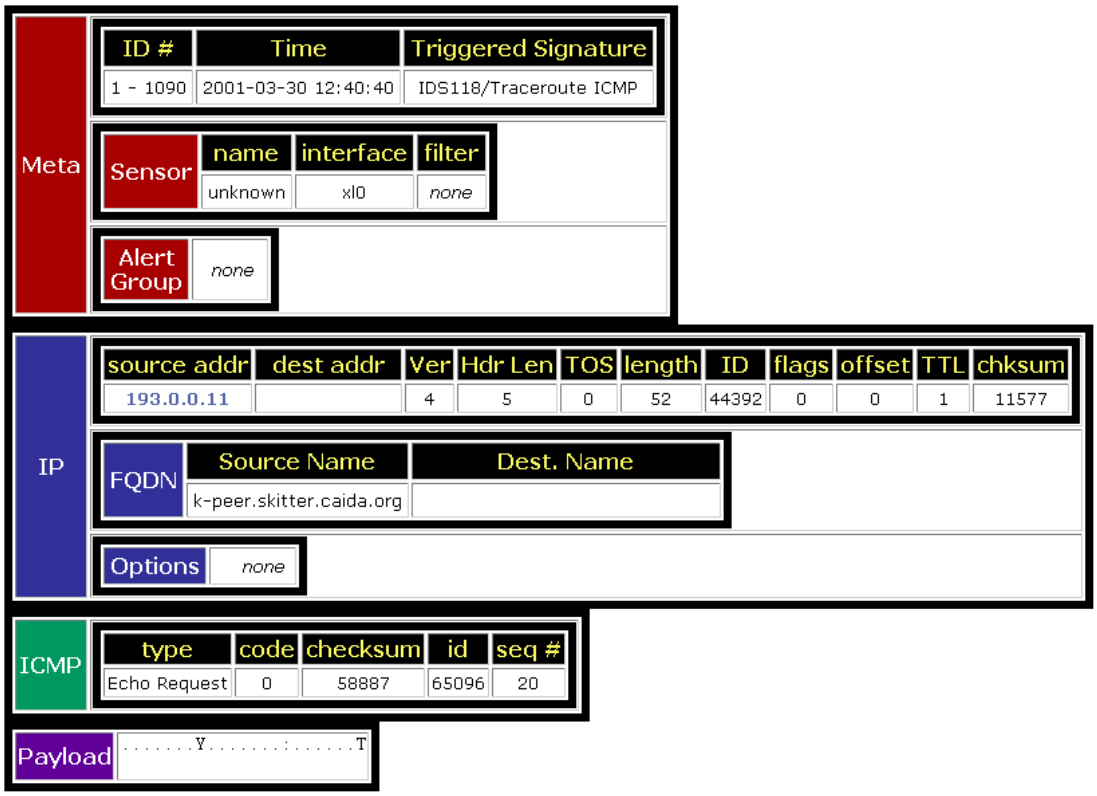
Feb 22 16:34:19 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.0.11-> w.x.y.z
Feb 22 19:15:00 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.0.11-> w.x.y.z
[222 alerts omitted]
Mar 30 09:36:03 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.0.11-> w.x.y.z
Mar 30 12:40:41 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.0.11-> w.x.y.z

Snort alerts – Part Two:

Feb 22 17:26:59 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.14.253 -> w.x.y.z
Feb 22 20:26:23 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.14.253 -> w.x.y.z
[229 alerts omitted]
Mar 30 10:05:32 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.14.253 -> w.x.y.z
Mar 30 12:59:44 ids1.target.co.nz snort: IDS118/Traceroute ICMP:
193.0.14.253 -> w.x.y.z

The above source addresses resolve to: k-peer.skitter.caida.org and k-root.skitter.caida.org.

The following image is taken from the acid database view of the Snort data:



This is the last alert shown in Snort Alerts – Part One above. The length of the packet is shown as 52 bytes.

Possibility the source address was spoofed:

Possible, this is UDP traffic, the source does not necessarily need to receive a response . However it was subsequently discovered that it is not spoofed

Description of Attack:

A significant number of Traceroute packets to the external DNS server. These amount to approx 450 alerts over a five-week period. Whilst this value is not sufficient to present a denial of service attack it is persistent.

Further investigation (once the cause was identified) revealed a further five servers from the same domain. These five servers generated an extra 1166 alerts during this period.

Each of these servers belongs to the netblock owned by Caida.

whois -h whois.crsnic.net caida.org

Registrant: CAIDA (CAIDA-DOM) UC, San Diego, La Jolla, CA 92093, US

It was determined (via http://www.caida.org) that Caida are using a tool known as 'skitter'. Definition from the web site:

"skitter is a tool for actively probing the Internet in order to analyze topology and performance."

Attack Mechanism:

This is stimulus traffic. This is not an attack, it is benign[2]. It is network performance analysis.

An abridged description of the mechanism comes from the Caida web site:

"Measure Forward IP Paths
skitter records each hop from a source to many destinations. by incrementing the 'time to live' (TTL) of each IP packet header and recording replies from each router (or hop) leading to the destination host."

Correlations:

Confirmation from skitter-configs@caida.org (the administrative email address cited on the web site), that this server is included in the database of systems that skitter analyses. The author gave permission for the inclusion of this email.

From: <tech support> [mailto: ___@ipn.caida.org]
Cc: 'skitter-configs@caida.org'
Subject: Re: confirmation

Sir,

[…]
you are on two of our active destination lists.
These two lists are on a total of about 10 monitors. Hope this
helps, and if you do decide to request removal, just let me
know and I can remove you promptly.

Thanks, and sorry for any inconvenience,

The web site also confirms that the packet size is 52 bytes.

Evidence of active targeting:

Targeted. A technique to determine network paths and speed to New Zealand (and elsewhere). The email above confirms targeting.

Severity:

Target Criticality = 4.
This is one of the external DNS servers

Attack Lethality = 1
This is benign external network analysis.

System Countermeasures = 1
This is a DNS server. It has to receive UDP to port 53 to function.

[2] Some would suggest that although benign, it is perhaps undesirable.

Network Countermeasures = 2
The Firewall allows UDP packets to the DNS server (it has to).
IDS threshold for this alert is set too high.

Attack Severity = **2** (4+1) – (1 +2)

Defence Recommendations:

Tune out traceroute alerts from these sources from the IDS rulesbase. Use post-processing to prevent these alerts from reaching Intrusion Analysts. Optionally ask to be removed from the Caida database and/or block the sources at the external router.

Question

Why would post-processing be recommended to remove/reduce these alerts?

A) Post-processing will prevent the source from determining path information
B) The Intrusion Analysts have enough false positives to process without these distracting alerts as well
C) Post-processing will cause reject packets to be sent to the source, reducing the TCP timeout and increasing bandwidth availability
D) All of the above

Answer: B. Post processing allows the number of false positives to be reduced and allows more sophisticated processing of the dataset. This reduces the load on the Intrusion Analyst.

5 Network Analysis Five

Source of Trace:

Home network, with ADSL modem (performing address translation) and an external Check Point Firewall-1 4.1 SP3 system (providing address translation).

An Internal Firewall (running a different firewall product) protects internal systems.

External and Internal IDS are deployed

Detect was generated by:

Snort Intrusion Detection system. Snort was running the standard rule set. The rule set has been tuned to suit the local environment. Supporting Tcpdump traffic is also provided.

The format of the IDS alerts below is: Date & Time, Snort-Sensor-Name[process ID], IDS Description, Source IP Address & port, Destination IP Address and port

Apr 8 12:41:33 exterior snort[1245]: spp_http_decode: IIS Unicode attack detected: 207.213.220.70:1296 -> 192.168.1.3:80

This traffic was also seen and logged by the Apache Web server.

The format of the Apache log below is: Source IP Address, Date & Time, request string from attacker, Return code. In this case the return code is 401 Unauthorised[3] as the web server concerned requires a password to enter.

207.213.220.70 - - [09/Apr/2001:02:47:55 +1200] "GET /scripts/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af/winnt/system32/cmd.exe?/c%20dir HTTP/1.0" **401** 397

Possibility the source address was spoofed:

The source address could be spoofed but this in unlikely. The IIS Unicode exploit requires the TCP three-way-handshake to complete.

The address resolves to: dhcp-207-213-220-70.kola.net. This appears to be a dynamically allocated address within the kola.net address range. Whois information is provided below:

> whois -h whois.arin.net 207-213-220-70
>
> Pacific Bell Internet Services, Inc. (NETBLK-PBI-NET-3)
> Marathon Plaza, North Tower, 303 Second St, Suite 830
> San Francisco, CA 94107, US

[3] RFC 2617 : The 401 (Unauthorized) response message is used by an origin server to challenge the authorization of a user agent. This response MUST include a WWW-Authenticate header field containing at least one challenge applicable to the requested resource

Netname: PBI-NET-3
Netblock: 207.212.0.0 - 207.215.255.255

Description of Attack:

Attempted exploit of IIS weakness in Unicode parsing.

Attack Mechanism:

This is stimulus traffic. The service targeted is the Web server. If the target web server were an unpatched Microsoft Internet Information Server, then the system could be compromised. This is an attempted exploit.

Correlations:

A number of individuals in NZ ADSL space provided corroboration. The following email is extracted from the NZ ADSL email list (with the author's permission):

From: Fran [mailto:fran@mobilecomputing.co.nz]
Sent: Sunday, April 08, 2001 11:45 AM
To: adsl@unixathome.org
Subject: Someone knocking on my door

dhcp-207-213-220-70.kola.net - - [08/Apr/2001:11:22:43 +1200] "GET /scripts/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c%20dir HTTP/1.0" **404**[4] 329

Fran
:):):)

This message is part of the NZ ADSL mailing list.
see http://unixathome.org/adsl/ for archives, FAQ,
and various documents.

Note the return code in this corroborating data is different. The return code is 404 Not Found. This is because although the Web server involved did not request authentication, it did not have the URI requested (cmd.exe is not commonly found on Linux / Unix platforms).

Evidence of active targeting:

The source was searching for IIS web servers with the Unicode vulnerability. No effort had been made to restrict the systems tested to those known to be running IIS. The scan traversed at least some of NZ ADSL space.

Severity:

Target Criticality = 2.
This is a non-critical web server

Attack Lethality = 2.
This is an attack against a Microsoft IIS server, used against a Linux Apache server.

[4] RFC 2616: 404 (Not Found): The server has not found anything matching the request-URI.

System Countermeasures = 2.
The web server does not have any known vulnerability to IIS Unicode exploits.

Network Countermeasures = 3.
The external firewall provides access to the web server on port 80. As a result the exploit will traverse the firewall to the web server. The internal firewall does not allow any traffic from the web server to reach the internal network.

Attack Severity = -1 (2+2) – (2+3)

Defence Recommendations:

This attack was targeted at IIS servers, and was ineffective on the target web server. Optionally block the source net and email the abuse email address for the netblock.

Question

Which of the following statements is true?

A) Apache web servers do not need to be patched because all known exploits target Microsoft IIS.
B) The firewall will normally prevent all compromise of the web server.
C) Any web server type may be vulnerable to attack, obscurity is not security.
D) All currently known IIS Unicode exploits are benign.

Answer: C. Whilst obscurity may improve security, it cannot be relied on. Strong layered defences are recommended for any systems connected to untrusted networks.

© SANS Institute 2000 - 2005 Author retains full rights.

Part Two: IDS Technology Paper

Network Intrusion Detection Systems: Deployment Techniques and Vulnerabilities

William Stallings states "Inevitably, the best intrusion prevention system will fail. A system's second line of defense is intrusion detection."

This paper discusses the role of Network Intrusion Detection Systems (NIDS or Network IDS) in the field of network security. A number of strategies and methods for improving the effectiveness of Intrusion Detection (ID) are provided.

The limitations of both the technology and the methodology of ID are discussed. Deployment recommendations are made for addressing these limitations.

A firewall can be considered to have failed if it allows traffic to traverse it when it should not. An IDS can be considered to have failed if it does not alert on an attack against a network it is monitoring. The firewall has the easier task.

Yet the IDS must be up to the task. As Marcus J. Ranum suggests "we will lash out in anger seeking retribution. The firewalls will be supplemented with tort lawyers and the IDS will become sources of evidence."

IDS systems are required that will effectively support the Firewall *and* be predictable and reliable enough to be used as forensic evidence in a court of law.

Network Intrusion Detection is a relatively recent development. This paper discusses the methods in which IDSs can be more effectively deployed, and seeks to highlight areas in which the IDS is still vulnerable to failure.

© SANS Institute 2000 - 2005 Author retains full rights.

1 IDS Techniques

There are two standard techniques for detecting intrusions: Anomaly Detection and Signature Based detection. Network IDSs commonly use either or both.

1.1 Anomaly Detection IDS Technique

Rebecca Bace describes anomaly detection as employing "statistical profiles of user behavior over time" to "characterize the behavior of systems."

At this time it is proving difficult to characterise the standard or expected behaviour of systems and networks. The behaviour of a single system running a single, simple application with a known (and limited) number of possible interactions *can* be modelled effectively. As a result an IDS could use this model to detect anomalous behaviour. However when the complexity of the applications used (and the operating systems on which these applications run) is considered then the characterisation becomes difficult if not impossible.

Consider a publicly accessible web server with a database back-end. The server might have thousands of different users every hour. Each of these users may interact with the server in a different manner. Some of these users will never have visited this web server before. Some may have persistent (and possibly free-form) data stored in the database. It is not practical at this time to characterise this behaviour.

Ranum argues that despite the fact that "much research is still being done […] these [anomaly detection] systems are not the answer."

1.2 Signature Based IDS Technique

IDSs commonly use databases of signatures representing known attacks, scans and probes.

Whilst signature based IDS have some disadvantages, as discussed in this paper, it is the chosen technique at this time. It is also the method employed by most free (e.g. snort: http://www.snort.org) and commercial (e.g. Internet Security Scanner: http://www.iss.net) IDSs.

1.3 Technique Chosen: Signature Based

The remainder of this paper discusses the use of Signature Based IDS techniques.

2 IDS Types: Network and Host based

There are two common types of intrusion detection systems available at this time: Network based IDS and Host based IDS. Both have problems. As Bruce Schneier asserts "IDSs are really still in their infancy, and different ideas are vying for supremacy."

Network based IDS can be thought of as a secret wiretap. The Network IDS watches all (ideally) the traffic on a network without announcing its presence.

Host based IDS systems run on a target server itself. The Host IDS monitors the applications on that server (for example a Web server) looking for local signs of intrusion. This may involve a number of techniques including checking the integrity and performance of the application.

This paper focuses on the limitations of Network based IDSs.

3 Signature Based Network IDS : Problems and Solutions

3.1 Network IDS Problems

The problems associated with Network IDSs are described below. They include false positives where the IDS triggers an alert on traffic that is in fact benign. They also include false negatives where the IDS fails to alert (or that alert is stopped or lost en route). Other problems discussed include IDS evasion or overload, encryption and packet manipulation.

3.2 Excessive overload of alerts

Problem

If the IDS system is not configured to automatically respond to an alert, then a human must process the alert. If the IDS system if generating excessive false positives (especially if these occur in the middle of the night) then there is significant danger that they will be ignored, or that their importance will be downgraded. This is human nature. If the IDS then triggers an alert on a similar genuine attack or break-in then that alert is likely to be ignored also.

Solution

The solution to this problem has two components: improved signatures and procedures to control the human involvement.

The signature databases (and the IDS itself) should be tuned to reflect the environment that it is protecting. If the Web server in use is based on Apache (http://www.apache.org) then the IDS does not need to alert[5] on Microsoft IIS (http://www.microsoft.com/iis) based attacks and vice-versa.

There are problems with this approach if a new server is introduced onto the LAN with a different type of Web (for example) server. The IDS will be blind to attacks on this new server. This approach can also increase the effort and time required to merge in newly released IDS rule sets.

Procedures should be established so that the human involvement is controlled and monitored, to prevent alerts being ignored. This is a non-trivial task.

3.3 Hidden Attack

[5] The IIS attacks should still be logged even though they are not being alerted on. The filtering process should usually occur after the IIS alerts have been logged so subsequent analysis processes have complete data.

Problem

This is related to the excessive overload issue discussed above. If a genuine attack is hidden (or couched) in the background noise of lots of benign attacks or false positives, it may not be noticed. There are tools available to manually (hping: http://www.eaglenet.org/antirez/hping2.html) and automatically (stick: http://www.eurocompton.net/stick and snot: http://www.geocities.com/sniph00) deliberately generate traffic that the IDS will trigger on.

Solution

A partial solution to this problem is to tune the IDS database to reflect the local environment and to ensure the IDS does not alert on attacks that would fail in the local environment (e.g. an Apache attack on a IIS server).

Despite this tuning, it is relatively simple for an attacker to identify a given web server type (the command string "HEAD / HTTP/1.0" will usually provide the Web server type and often the Operating System it is running on[6]). As a result the tuning may not provide much assistance. It is possible to proactively block source addresses from an IDS. In this circumstance an internal IDS could be linked to a Firewall to automatically block source addresses. Unfortunately this can often become a Denial of Service on the business itself (or its partners if the address is spoofed – see the following section).

There is significant benefit in providing an analysis engine that uses thresholds. The engine could look for a certain number of alerts from the same source address within a given time period. Nevertheless it is possible to evade this by sending the noise from address A and the attacks from a different address B.

3.4 Spoof Partners Address

Problem

If an 'attack' uses the spoofed source address of a business partner then it is possible that an automated IDS system might then block access from that valid business partner. This results in a Denial of Service against genuine business partners.

Solution

The IDS should know (i.e. be told) the network addresses of all business partners and trusted or semi-trusted networks. There should be additional (probably human) checks before automated blocking is performed on such networks. Authenticated protocols (IPsec Authenticated Headers and similar can also assist in this area)

3.5 Denial Of Service & Fail Open

[6] Both of these strings can be modified to obfuscate the server type. This is not commonly done however.

Problem

An IDS can be subjected to a denial of service attack. It is a relatively simple task to identify what traffic an IDS will trigger on (since they all use a core set of signatures). Sending known recent attacks against the appropriate server types (i.e. IIS vulnerabilities against an IIS server) requires reconnaissance but increases the probability that the IDS will alert. The IDS can be overloaded by being sent a large quantity of traffic that the attacker knows it will trigger on. It may also be possible to send traffic that will cause the IDS (or the operating system it is running on) to fail.

Unlike most firewalls, IDSs fail-open. When a Firewall fails, traffic is usually no longer passed through that firewall. When an IDS fails the traffic is no longer monitored, but can continue to use that network. The IDS is nullified.

Solution

This solution requires multiple IDS sensors. A sensor should be located outside the firewall. One or more sensors should be placed inside the firewall.

It is accepted that the external IDS sensor will see all the traffic inbound to the firewall. The internal sensor should see a subset of the traffic, since the firewall will have filtered it.

Ensure that the IDS hardware is of a high-specification. It is necessary to test the IDS, with a recent rule set, using a tool such as stick[7]. This will verify that the IDS does not drop packets nor consume all CPU and disk resources.

Provide a mechanism (heart-beat) to ensure the IDS is alive and working.

Using different IDS solutions and vendors decreases the probability that all of them will be vulnerable to the same denial of service or attack.

3.6 Encryption

Problem

Many servers now support encryption to provide confidentiality over untrusted networks. This encryption might take the form of a Virtual Private Network (using PPTP or IPsec etc.) or a Secure Sockets Layer (SSL) session with a Web server.
In these circumstances the data of the traffic is not easily readable by the IDS. Whilst the sensor may still be able to read (and react to) the packet headers it cannot look at the payload to perform the signature matching.

Web servers that support both encrypted and non-encrypted sessions are now being attacked via the secure SSL method, as attackers can expect to avoid detection.

[7] See section 3.3

Solution

There are no simple solutions to this problem. The IDS could have the encryption keys to decode the traffic[8]. This would reduce the level of security as the IDS is not designed to store secret encryption keys and any *increase* in the number of locations where the keys are stored is a *decrease* in the security thereof). The target server could provide either the key or a decrypted version of the payload via an out-of-band mechanism.

Another technique (for web traffic) is to use a dedicated front-end system to decrypt the traffic and then pass the data to the web server in a reverse proxy manner. The IDS can then examine the unencrypted traffic.

This area remains an issue for Network IDSs. Indeed, to address this the Network IDS needs to assume some of the properties of the Host based IDS.

3.7 Continuous update required

Problem

Like Anti-virus systems that cannot identify viruses that are not defined in their signature files, IDS systems cannot alert on traffic they are not instructed to alert on. As a result it is necessary to ensure that signature files are kept up to date at all times.

Solution

To a certain extent this process can be automated. There are methods to automate the update of IDS rules and signature files on a regular basis. IDS administrators can often handcraft IDS rules to reflect new security concerns.

Nevertheless, as with anti-virus signature file updates, this is a reactive process, there will always be a window of vulnerability between the creation of an effective attack and the subsequent signature file update (and its widespread deployment in the field).

3.8 Signature Mutation

Problem

Many signature files used with IDS systems include patterns that reflect some unique aspect of the payload or TCP header. For example IDS 430 (Bugtraq ID1786) has the string "?STRENGUR “ and IDS 398 (CAN 1999-0660 for CVE) has a standard port of 31337[9]. The recent Lion attack on Linux LPR port 515 includes the following in the payload “1i0nip@china.com”.

---

[8] This is only effective when encrypting using shared secrets. If a key exchange technology is in use (e.g. IKE) this will not work.
[9] The familiar ‘misspelt’ elite.

Many of the creators of such attacks like to 'sign' their work. This often involves including some convolution of their name (or tag) in the payload of the packets associated with the attack. This makes it much easier for the signature database to be constructed.

A competent attacker can mutate the attacks so that they bypass the IDS sensor. It is possible to change the string above to "?STRONGUR " or perhaps modify the port number for the attack. The IDS might then not alert on these strings. A mutated lion attack might send email to t1g3r@country.com instead, rendering the signature ineffective.

Solution

There is no obvious solution to this problem. The string matching technology used could be allowed to permit (or try) permutations of the signature strings, but this requires time and computing power. This approach would also increase the number of false positives generated.

Modifying the port of a Trojan might reduce its effectiveness (as the clients would be listening on a different port) perhaps even until the signature database can be updated to reflect the modification.

3.9 Network Segmentation: Switches

Problem

Many networks are now segregated by switches (and VLANs). These switches offer significant benefits to network throughput.

The nature of a switch is such that the IDS will not normally see the data that is sent to the target server (as they are on different switch ports)

Solution

Whilst there are methods to mirror data between ports on a switch, there are issues with the speed and functionality of these methods.

A preferred solution is to place the IDS and the target server in a pair on a small repeated network on a single switch port. This solution ensures that both systems see the same network traffic. This solution does not scale well to larger networks.

3.10 Packet Manipulation

Problem

There are a number of ways in which a malicious attacker can craft packets to manipulate the IDS. A common technique is to reduce the Time-to-live (TTL) value of the packet header such that either only the IDS or only the target server see the

packet.

An alternative approach is to manipulate the packet size and Don't Fragment (DF) flag such that it cannot traverse a network with a smaller MTU[10] where (for example) the IDS is located. If such a packet does reach a network with a larger MTU where a target server is located, then the IDS is blind to an attack on that server.

Other techniques involve changing the packet length and data size.

Solution

The IDS sensor should be located on the same physical network segment as the target server. Any traffic that reaches the sensor will also reach the target server and vice-versa.

3.11 Unicode Evasion

Problem

A recent attack on Web servers has been the exploit of Unicode[11] character translation. Unicode allows a much larger character set than standard ASCII and uses pairs of characters to represent single characters. Unfortunately the translation scheme defined is not implemented in a consistent manner across all web servers and operating systems.

This allows the manipulation of the data stream to use Unicode to avoid the patterns in the IDS signature database.

Recent tools such as whisker produced by rain forest puppy[12] have options to attack web servers in one of ten different 'IDS evasive modes'. Some of these modes employ Unicode techniques.

Solution

The simple solution is to disable Unicode on the server. This solution will only work for servers that do not require Unicode. For servers that require Unicode this does not solve the problem.

Providing Unicode translators to parse the Unicode stream as it passes can alleviate this problem. This decode will need to be consistent with the decode at the target server. This is another reason for using the same platform for the sensor as for the server.

Another approach would force a Web server (for example) to decode the Unicode and then seek approval from the IDS before performing the action specified by the

[10] The Maximum Transmission Unit (MTU) is a measurement of the maximum size of packet that the network will support. For Ethernet this is usually 1500.
[11] Also referred to as UTF-8. Refer to http://www.unicode.org for further information.
[12] http://www.wiretrip.net/rfp/

decoded string.

This latter approach requires an out-of-band interaction between the Web server and the IDS. There is significant overlap here with host-based IDSs.

3.12 Resource Starvation/Exhaustion

Problem

The IDS must read all the packets on the wire to perform its analysis. This is normally more traffic than any given host it is protecting will receive. The IDS must also (if configured to do so) perform fragmentation reassembly (which is memory intensive, especially if the fragmentation is malicious in nature).

The speed of networks is such that all IDS systems will drop packets at some point. The bandwidth that is available will at some point exceed the ability of the IDS to monitor every packet on the wire.

The IDS must also store its log files and TCP packet dumps. This can consume significant amounts of disk space.

Solution

A partial solution to this resource exhaustion is suggested and consists of a number of different aspects.

The IDS hardware platform must exceed the performance of that of the target servers it is protecting. Testing must be performed to identify the maximum sustained performance of the IDS and the Firewall or perimeter device should throttle the dataflow to below that level.

Networks should be subdivided where possible to pair IDS systems with single hosts that they are protecting. This solution does not scale well.

3.13 Differences Between IDS and Target Server

Problem

The IDS system (sensor) in use is likely to be different (often very different) from the system that it is monitoring. This provides the opportunity to break one system without breaking the other.

These differences might extend to hardware type (for example the NOP[13] command on an Intel platform is 0x90 whereas on an SGI platform it is 03 e0 f8 25 – this might

[13] NOP – No Operation command to the CPU. This is commonly used within buffer overflow attacks.

defeat simplistic NOP scanning).

These differences might also include different operating systems (say Unix or Windows NT) and different TCP/IP stacks. Each of these differences can be exploited by an attacker and so reduce the effectiveness of the IDS.

Solution

There are some partial solutions to this problem. It is possible to arrange the network such that each target server has a dedicated IDS sensor. This IDS sensor can be built on the same platform type as the server it is protecting. The sensor can be constructed to use the same operating system (and version, hotfix etc) as the target server.

This approach may severely limit the type of IDS that is used. Indeed it may also limit the type of target server used.

This approach of pairing IDS systems with targets also allows much finer tuning of the sensor to reduce false positives as the rules can reflect a single known host.

There are disadvantages to this approach that must be recognised:

- Extra IDS sensors increase the cost
- Extra IDS sensors increase the management and maintenance costs
- Any effective compromise against the platform could take out both server and sensor

3.14 Inherent Failures

Problem

Any given platform, operating system or intrusion sensor will have inherent problems. These may include hardware failure, software bugs or design errors

Solution

The solution to this problem involves implementing multiple sensors from different vendors. A deficiency in one IDS type is unlikely to be repeated in a different type (although it is possible). The overheads of multiple sensors are discussed in section 3.13.

4 Conclusion

4.1 Overview

This paper has provided some potential solutions to problems associated with the deployment of Network based Intrusion Detection Systems.

Common problems with Network IDSs have been highlighted and solutions proposed. It is accepted that some of the solutions are only partial and that there are still some areas in which Network IDS systems are only partially effective.

4.2 Problems Solved

This paper has identified a series of problems relating to the effectiveness of Network IDS as part of the security infrastructure.

The problems identified for which solutions have been proposed are given below:

- Excessive overload of alerts
- Couched attacks
- Denial of service and fail open
- Spoof partners address
- Network segmentation: switches
- Packet Manipulation
- Resource Starvation/Exhaustion
- Unicode evasion (partial solution)
- Differences between IDS and target server (partial solution)

4.2.1 Problems Outstanding

It is accepted that there are still problems for which there are not yet reliable or practical solutions, these are listed below:

- Encryption
- Continuous update
- Signature Mutation

4.3 Perimeter Defences

There is substantial benefit in providing effective perimeter protection. It is suggested that a stateful perimeter router can remove much of the false positive generating traffic

that an IDS outside the Firewall sees.
It is also recommended that this perimeter router be capable of, and configured to, drop all illegal packets. There is little advantage in this router allowing (for example) SYN-FIN packets to transit through to the Firewall and IDS.

4.4 The Perfect IDS

There is not likely to be a perfect IDS. Just as there is unlikely to be a perfect Firewall (although a severed network cable comes close). Nevertheless, there is considerable value in deploying Intrusion Detection within the network, in conjunction with other defences.

Ranum again: "the [IDS] technology should only be used as part of an overall strategy – and where it will be the most effective, not where it will generate the most hits."

References

[1] Stallings, William. Cryptography and Network Security. Prentice Hall, ISBN 0138690170. 1998. 490.

[2] Ranum, Marcus J. "Buffer Overruns and burglar alarms." Net Police Blotter, Issue #2. August 2000. URL: http://pubweb.nfr.net/~mjr/usenix/index.shtml (30 April, 2001)

[3] Bace, Rebecca Gurley. Intrusion Detection. Pearson Higher Education, ISBN 1578701856. 2000. 87.

[4] Ranum, Marcus J. "Is Intrusion Detection Software Being Used Correctly." July 1998. URL: http://www.securitymanagement.com/library/000556.html (28 April, 2001)

[5] Schneier, Bruce. Secrets & Lies. John Wiley & Sons, ISBN 0471253111. 2000. 196.

Part Three: Analyse This

Executive Synopsis

Introduction

GIAC Enterprises (GIAC) requested the assistance of eeek! Security Consultants (eeek!) to perform analysis of their IDS data for an eight-week period between 24th November, 2000 and 18th January, 2001. This report presents the results of that analysis to GIAC.

eeek! would like to thank GIAC for this opportunity to provide Intrusion Detection Analysis services. We believe the information contained within this report raises a number of issues with respect to the Security Infrastructure in use at GIAC, and we would welcome the opportunity to assist GIAC in implementing the remedial recommendations provided in this report.

Confidentiality

This report contains information that would be of substantial value to malicious agents. It is recommended that its distribution be restricted.

Audience

This report is intended for GIAC personal who are familiar with TCP/IP networking, intrusion detection and network security.

Network Security

The current level of network security is insufficient. Whilst there appears to be an effective deployment of Intrusion Detection Systems (IDS), the data from this IDS is not fully available. It is not apparent whether the lack of complete data is due to benign or malicious causes.

The IDS rule sets in use are not tuned to reflect GIAC's network and generate significant false positive data.

The firewall appears to overly permissive. eeek!'s Best Practice dictates a firewall policy of 'Default Deny'. Default Deny decrees that any traffic not explicitly permitted is denied. GIAC's firewall deployment does not appear to follow this principle.

Compromised Hosts

A significant number of hosts may have been compromised. They should be removed from the network and forensic analysis should be performed. The hosts are identified within this report. At least one virus (or worm) has been targeted at an internal system. Anti-virus measures should be reviewed. Gateway content scanning is

recommended.

Ongoing Attack

There is significant evidence in the log files provided to suggest that GIAC's network is under ongoing attack. It is also being scanned repeatedly (though this is common on public networks). Over 7,500 scans were recorded in the two-month period provided. External address spoofing[14] is apparent, as is packet crafting[15].

GIAC Security Focus

The scan and attack distribution showed that the highest amount of activity was concentrated around the four weeks including Xmas & New Year and early January. This is to be expected, as potential attackers are less likely to be spotted by a reduced number of security personnel. It is recommended that GIAC does not reduce its security focus at these times of the year.

GIAC provided in the order of 20 million lines of IDS data for a two-month period. It is suggested that IDS data should be analysed on a more regular basis. Two months provides a substantial window for attack without reaction, and produces an excessive amount of data.

Initial Recommendations

Whilst it is inappropriate to make formal recommendations to GIAC regarding blocking external hosts without further investigation, the following are suggested as areas to address in the short term.

- Block the Watchlist source address ranges (159.226.0.0 and 212.179.0.0).
- Drop all malformed (e.g. SYN-FIN) packets at the Firewall / perimeter router.
- Block outgoing access to port 515 until further investigation is completed
- Block incoming access to port 515 until further investigation is completed
- Drop ping traffic to broadcast addresses
- Block externally sourced packets for port 1080 (requires stateful Firewall)
- Drop externally sourced packets to port 31337
- Drop externally sourced packets to the Netbios port range (137-139)
- Drop externally sourced packets to RPC (port 111, aka portmapper)
- Contact the ISPs associated with addresses attacking (or scanning) GIAC.
- Block SNMP traffic at the Firewall or perimeter router.
- Deploy additional IDS sensors in the network.

---

[14] TCP traffic that appears to have originated from address A when in fact it came from address B. IP addresses include 4.4.4.4 and 8.8.8.8 in consecutive alerts, whilst possible, it is unlikely that the traffic originated from these addresses.

[15] Packets are included in the logs that are not legal TCP packets. The flags are not set correctly or contradict each other.

1 Detailed Timeframe Information

Timeframe

This report details the analysis of IDS data provided by GIAC to eeek! The data covers the periods listed in the table below:

Data Type	Earliest Data	Latest Data	Number of Logs	Number of Days	Percentage
Alert (this is Snort[16] IDS alert data in a text format)	24th November 2000	18th January 2001	44	56	79%
Scan (this is port scan and TCP flag information)	5th December 2000	15th January 2001	27	32	84%
OOS (this is malformed TCP packet information)	28th November 2000	18th January 2001	22	52	42%

As can be seen from the above table significant portions of data are not available for the period concerned.

Twelve days of alert logs were not provided. As a result there were 12 days in which attacks could be purposely perpetrated without detection. It is also possible that a malicious agent deleted these alert files (or filled the disk to cause their loss).

The lack of complete data for the scan and Out-of-sync (OOS) files is considered less critical but till represents a period for which analysis cannot be performed.

[16] A commonly used and free Intrusion detection system See http://www.snort.org for more information.

2 Overall comments

2.1 Intrusion Detection Systems

Intrusion detection systems provide a supporting role in network security. They should be used in conjunction with Firewalls and additional security systems. There is a significant advantage in placing IDS sensors both inside and outside the network perimeter. IDS logs should be reviewed in conjunction with Firewall logs and system audit trails.

2.2 Limitations of Tools used

Snort

Snort is a commonly used free intrusion detection system. It is commonly deployed on Linux platforms. Whilst Snort has certainly increased the use of IDS around the Internet, there are some issues that need to be considered from a deployment perspective.

Snort is, in common with most other IDS systems, primarily a packet matching IDS. It looks for packets on the network with signatures that it recognises and alerts when it recognises such signatures.

The use of pattern matching in this manner is suspect to both false positives and false negatives. These are discussed below.

False Positives

False positives, in this context, occur when the IDS triggers an alert when identifies a traffic pattern as matching a signature when the traffic is not actually indicative of the attack (or scan) associated with that pattern.

A (simplified) example might be the IDS detecting the NOP operation in a packet. NOPs are (in an Intel environment) identified by the hex value 0x90. A series of NOPs in the data of a packet can be indicative of a buffer overflow. In the case of a false positive, the 0x90 values might just be part of the Web page being viewed.

It should be noted that the analysis provided has included alerts that may well be false positives. Without more precise information (i.e. sensor location, system identification) it is not possible to refine these out of the assessment. It is also considered preferable, in this post-activity analysis, to include false positives rather than inadvertently exclude real positives.

False Negatives

False negatives, in this context, occur when the IDS fails to identify a real attack. Whilst there any many causes for this, the most common cause is that the IDS was

not told to look for the traffic signature associated with the attack or exploit.

In this manner the IDS system is akin to an anti-virus scanner that does not have the signature for a new virus.

Tuning

A standard Snort deployment is likely to come with a default set of signatures. Newer version of such signatures can be downloaded, in a similar manner to updated anti-virus signature files.

There is a need to tune the Snort signature set (or rules base) to suit the local network environment. The standard rules set is likely to generate significant false positives.

It should be noted that on the busiest day for IDS alerts, January 7th, 2001, there were over 20,000 alerts. This is an excessive amount and represents a significant workload for an analysis team in one day. Tuning will reduce this quantity (as will tightening the Firewall rules base).

eeek! can assist GIAC in this tuning process.

2.3 Application of IDS tools at GIAC

As part of this assessment, we would like to provide some feedback to GIAC regarding their use of Snort and the quality of the data provided to ourselves.

Snort Rules base

The Snort rules base in use is described as a 'fairly standard' rules base. There would be some advantage from an analysis perspective if eeek! were able to view the exact rules base in use at the time the data was generated.

It is suggested that the Whitehats rules base (http://www.whitehats.com), whilst not as extensive as the standard one (http://www.snort.org), provides a better-supported rule set. The Whitehats rules base is also subject to greater sanity checks than the standard rule set. It also cross-references into the Whitehats database and the Common Vulnerabilities and Exposures (CVE) database.

Tuning

The effectiveness of an IDS is directly related to the level of tuning of the rules set. Tuning a rule set enables the IDS to focus on traffic that is more likely to be malicious. The tuning exercise should also remove a lot of the noise that is present in the data provided to eeek!

There is a need to perform this tuning at GIAC. eeek! can assist GIAC in this activity.

Missing Data

As has already been advised, there are substantial data records missing from the period eeek! has been analysing. It is strongly recommended that GIAC implement

mechanisms to ensure the storage and uninterrupted availability of such data.

There is a possibility that external parties have deliberately removed malicious behaviour that was detected by the IDS. As such the integrity of the remaining data is suspect.

TCPdump Information

Snort is able to read data in a TCPdump[17] formatted log file. It would have been advantageous if GIAC could have provided such a file, in addition to the text log files. This would have allowed packet replay. GIAC should consider retaining and providing such TCPdump files.

IDS Log File Assessment

It is recommended that GIAC assess (with eeek!'s assistance if required) their IDS log files on an ongoing basis. It is preferable to assess these logs daily rather than monthly. There were over 24,000 internal systems represented in the data provided. Security Best Practices would dictate a more frequent analysis of a small dataset.

eeek! can provide intrusion analysis training to GIAC personnel if required.

Data Provision

It is recommended that the filename scheme used at GIAC be modified to more appropriately reflect the actual data. Filenames could reflect the date of the data contained therein. This will make it easier to track the log files. It should also be noted that there was duplicate data in files with different names provided to eeek!

The data was provided in a clear text, unencrypted form. The data contained information (address ranges, potential compromised hosts, IDS system, effective attacks) that would be of substantial value to malicious agents. It is recommended that such data be encrypted before transfer over untrusted networks.

Missing Information

It would have been valuable if GIAC had been able to provide the following information as was requested:

- Network Topology diagram
- IDS sensor location information
- Server identification and IP address information
- Security policy and defence configuration

2.4 Analysis Process

[17] TCPdump is a standard Unix and Linux utility to capture TCP packets.

Data Provision

The data was provided as a single archive. The archive was extracted and separated into weekly directories. Missing data was identified.

There are periods where there is limited or no data available. This reduces the ability to corroborate assessments. It is possible that additional attacks and potential host compromises occurred during such periods.

The data was in three formats: alert data, portscan data and Tcpdump data showing out-of-specification packets.

Initial Data Manipulation

A series of unix command line tools including 'awk', 'lex', 'cut' and 'grep' were used to manipulate the data. These commands are effective at processing substantial amounts of data. It is suggested that Perl scripts would also be valuable in this area, but they were not used in this instance.

A simple lex pattern substitution was used to standardise the formatting for further manipulation. The 'lex' code fragment is shown below:

```
%%
\[\*\*] (printf("~"); }
-\> (printf("~"); }
from (printf("~"); }
```

Further substitutions replaced MY.NET with 10.1 to facilitate further processing.

The unix 'for' and 'seq' commands were used to construct a file with IP addresses and Hostname pairings. The command is shown below.

```
for i in `seq 0  255` ;
do
  for j in `seq 0 255` ;
  do
    echo 10.1.$i.$j  MY.NET.$i.$j
  done
done > /tmp/hosts
```

This produced a file thus:

```
10.1.0.0          MY.NET.0.0
10.1.0.1          MY.NET.0.1
10.1.0.2          MY.NET.0.2
…..
10.1.255.253    MY.NET.255.253
10.1.255.254    MY.NET.255.254
10.1.255.255    MY.NET.255.255
```

The output from this was then appended to the /etc/hosts file on the Linux analysis system to allow rapid name resolution.

The unix command 'awk' was also used to extract certain fields from the log files. Since the 'lex' commands above fixed the field separators as the tilde (~) character, the following command will print out the second field of the log file only. If this field is source addresses, then the following sequence will provide a sorted list of sources (and count them).

awk –F~ •{ print $2 }• logfile | sort | uniq –c | sort -n

Analysis Reporting : Cross Reference

Where possible, reference is made to the Common Vulnerabilities and Exposures (CVE) database provided via http://cve.mitre.org. This provides a standards-based database of vulnerabilities, exploits and attacks. This is intended as a consistent method for recording such activity.

Reference is also made to Whitehats (http://www.whitehats.com). This site provides the "advanced reference archive of current heuristics for network intrusion detection systems" (arachnids). This comprehensive database of network attack "signature" information can dynamically create and export signature strings that are compatible with free IDS software such as Snort. This signature database assigns an IDS number to each signature. These IDS numbers are referenced in this report.

3 Alert Assessment – 4 x 2 Week Periods

3.1 Time Period Breakdown

The eight-week period for which data has been provided has been divided into four two-week periods. This allows a more detailed breakdown of the activity seen without creating too much data.

Time Periods Covered

Period	Start Date	End Date	Number of Alerts
Fortnight 1	24th November 2000	7th December 2000	22,874
Fortnight 2	8th December 2000	21st December 2000	25,674
Fortnight 3	22nd December 2000	4th January 2001	52,019
Fortnight 4	5th January 2001	18th January 2001	89,964

Note that the Fortnight 3 period covers both the Xmas and New Year holidays. This is often a time of heightened attack as a result of reduced system administration and intrusion analysis resource availability (amongst others). January 7th was the busiest single day recording 21,306 alerts (this averages to an alert every four seconds!). The number of alerts is increasing throughout the period. Both the defences and the IDS strategy need revisiting to reflect this increase.

3.2 Fortnight 1 Alert Assessment

The following table provides the list of alerts that Snort generated in the period between 24th November and 7th December. It also provides information of the number of source hosts and destination hosts (both internal and external).

Alert Description	Number of Alerts	Number of Source Systems	Number of Destination Systems
Watchlist 000220 IL-ISDNNET-990517	18718	25	40
SYN-FIN scan!	1952	2	1915
Watchlist 000222 IL-ISDNNET-990517	732	13	12
WinGate 1080 Attempt	457	120	253

Attempted Sun RPC high port access	352	8	7
Queso fingerprint	227	12	18
Broadcast Ping to subnet 70	136	13	1
SUNRPC highport access!	72	7	5
SNMP public access	59	5	3
Null scan!	47	41	39
Back Orifice	36	3	36
Tiny Fragments - Possible Hostile Activity	25	6	6
NMAP TCP ping!	23	9	10
connect to 515 from outside	14	5	5
SMB Name Wildcard	13	2	2
connect to 515 from inside	5	4	4
External RPC call	3	2	2
Probable NMAP fingerprint attempt	2	2	2
site exec - Possible wu-ftpd exploit - GIAC000623	1	1	1

3.3 Fortnight 2 Alert Assessment

The following table provides the list of alerts that Snort generated in the period between 8th December and 21st December. It also provides information of the number of source hosts and destination hosts (both internal and external).

Taking the previous fortnight as a baseline, the following 'new' attacks were identified:

- Russia Dynamo - SANS Flash 28-jul-00
- TCP SMTP Source Port traffic

Alert Description	Number of Alerts	Number of Source Systems	Number of Destination Systems
Watchlist 000220 IL-ISDNNET-990517	11037	18	33
SYN-FIN scan!	8144	9	7446
connect to 515 from outside	422	3	2872
Attempted Sun RPC high port access	873	10	8
Russia Dynamo - SANS Flash 28-jul-00	546	2	2
WinGate 1080 Attempt	326	104	110
Tiny Fragments - Possible Hostile Activity	125	14	4
Watchlist 000222 NET-NCFC	123	14	11
SUNRPC highport access!	78	8	4
Null scan!	60	56	42
SMB Name Wildcard	31	17	14

SNMP public access	25	5	2
External RPC call	24	5	22
Queso fingerprint	24	7	13
NMAP TCP ping!	19	5	7
connect to 515 from inside	12	4	4
TCP SMTP Source Port traffic	3	2	2
site exec - Possible wu-ftpd exploit - GIAC000623	1	1	1
SITE EXEC - Possible wu-ftpd exploit - GIAC00023	1	1	1

3.4 Fortnight 3 Alert Assessment

The following table provides the list of alerts that Snort generated in the period between 22nd December and 4th January 2001. This covers the Xmas and New Year holiday period. A higher number of alerts are common at this time. The table below also provides information of the number of source hosts and destination hosts (both internal and external).

Taking the previous fortnight as a baseline, the following 'new' attacks were identified:

- Back Orifice
- Happy 99 Virus

Alert Description	Number of Alerts	Number of Source Systems	Number of Destination Systems
Watchlist 000220 IL-ISDNNET-990517	44998	13	20
Tiny Fragments - Possible Hostile Activity	3118	15	5
SYN-FIN scan!	1256	11	1249
WinGate 1080 Attempt	736	162	173
Null scan!	452	249	64
Watchlist 000222 NET-NCFC	385	17	11
Attempted Sun RPC high port access	307	4	4
SNMP public access	240	6	2
Queso fingerprint	100	6	6
NMAP TCP ping!	136	12	25
TCP SMTP Source Port traffic	93	2	2
SUNRPC highport access!	23	6	6
External RPC call	20	6	4
Broadcast Ping to subnet 70	12	8	1
Probable NMAP fingerprint attempt	4	1	2

Back Orifice	4	2	2
connect to 515 from inside	3	1	1
Happy 99 Virus	1	1	1
connect to 515 from outside	1	1	1

3.5 Fortnight 4 Alert Assessment

The following table provides the list of alerts that Snort generated in the period between 5th January and 18th January. It also provides information of the number of source hosts and destination hosts (both internal and external).

Taking the previous fortnight as a baseline, the following 'new' attacks were identified:

- DNS udp DoS attack described on unisog
- STATDX UDP attack

Alert Description	Number of Alerts	Number of Source Systems	Number of Destination Systems
SYN-FIN scan!	36788	14	24566
Watchlist 000220 IL-ISDNNET-990517	31164	13	21
DNS udp DoS attack described on unisog	16146	8	6
Tiny Fragments - Possible Hostile Activity	2068	16	5
Watchlist 000222 NET-NCFC	1152	13	15
WinGate 1080 Attempt	643	189	164
NMAP TCP ping!	376	31	146
SMB Name Wildcard	370	74	157
Queso Fingerprint	333	30	27
Null Scan!	266	195	65
SNMP public access	254	6	6
Attempted Sun RPC high port access	208	7	6
connect to 515 from inside	139	3	90
SUNRPC highport access!	31	6	6
External RPC call	12	2	3
Broadcast Ping to subnet 70	5	2	1
Back Orifice	5	4	4
Probable NMAP fingerprint attempt	2	2	2

connect to 515 from outside	1	1	1
STATDX UDP attack	1	1	1

4 Alert Assessment – Alert Types

4.1 Overview

This section discusses the alerts that were generated over the periods listed above. The following information is provided:

Information	Comment
Alert Description	A fuller explanation of the alert is provided
Common Vulnerabilities and Exposures	This is a list of identified vulnerabilities and signatures associated therewith
Whitehats IDS cross reference	This is a standard cross reference for Snort rules sets
Snort rule	Where appropriate, an example rule has been provided. This rule is similar to that on GIAC's Snort system that caused the alarm. The rules are adapted from those found at http://www.snort.org and http://www.whitehats.com
Trend	The number of alerts over the four 2-week periods is provided
Sources Identified	The source hosts or networks are provided where this provides useful information
Destinations Identified	The destination hosts or networks are provided where they provide useful information. In the case of scans this information is not often useful
Repeat Offenders	Where the source system appears in multiple alerts this information is provided
Summary	Where appropriate, a summary is given for the alert

4.2 Watchlist 000220 IL-ISDNNET-990517

Alert Overview

This alert is triggered as a result of the IDS sensing the Watchlist 000220 IL-ISDNNET-990517 signature. Further information on this attack is not currently available.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
18718	11037	44998	31164

This is the most prevalent alert over the two-month period

Sources Identified

The following sources were identified as the source of these alerts:

212.179.7.173	212.179.21.74	212.179.41.207	212.179.56.5
212.179.7.36	212.179.27.111	212.179.44.106	212.179.58.174
212.179.8.164	212.179.27.6	212.179.44.119	212.179.63.10
212.179.15.122	212.179.30.3	212.179.45.241	212.179.77.20
212.179.16.107	212.179.33.254	212.179.45.73	212.179.79.2
212.179.17.4	212.179.37.92	212.179.51.14	212.179.95.5

All of these IP addresses are associated with providers based in Israel (some addresses have been omitted for clarity. These addresses were within the same netblocks).

Repeat Offenders

It should be noted than none on the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers.

Summary

If there is no business need to communicate with these systems in Israel then access from the 212.79.x.x networks above should be blocked at the external router (or firewall).

If there is a need to communicate with systems in Israel it is recommended that such communication be restricted to designated hosts where possible. Extra monitoring is recommended. Target systems should be checked for vulnerabilities. Log analysis frequency should be increased.

4.3 SYN-FIN Scan

Alert Overview & Definition

This alert is triggered as a result of the IDS sensing packets with both SYN and FIN set[18]. This is an illegal TCP packet.

http://www.whitehats.com IDS198 : A TCP probe was sent with the SYN+FIN flags set in the header. This traffic does not occur naturally and indicates an intentional probe, likely as a part of single-packet OS detection.

[18] SYN is used to initiate a TCP conversation, FIN is used to terminate a TCP conversation. This is analogous to saying hello and goodbye in the same sentence without any words in between.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert TCP $EXTERNAL any -> $INTERNAL any (msg: "IDS198/SYN FIN Scan"; flags: SF;)

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
1952	8144	1256	36788

There is a significant increase in this alert in Fortnight 4.

Sources Identified

The following sources were identified as the source of these alerts:

24.113.198.51	cr859517-a.surrey1.bc.wave.home.com
63.11.25.117	1Cust117.tnt1.yakima.wa.da.uu.net)
63.204.152.253	adsl-63-204-152-253.dsl.snfc21.pacbell.net)
63.252.94.211	A050-0465.LAUR.splitrock.net
63.253.143.107	A040-1123.LAUR.splitrock.net
64.196.23.118	zzz-064196023118.splitrock.net
64.196.112.164	A010-0164.WLDF.splitrock.net
139.130.61.206	<no DNS name available>
209.221.206.188	<no DNS name available>
209.255.180.197	A010-0451.LAUR.splitrock.net
209.255.215.87	A040-0595.PHL2.splitrock.net
209.255.214.63	A040-0317.PHL2.splitrock.net
213.76.100.162	pe162.warszawa.cvx.ppp.tpnet.pl

Destinations Identified

The target systems number approximately 2000 spread throughout the internal network, this is consistent with a reconnaissance scan.

Example Data

12/28-20:16:28.950055 63.204.152.253:53 -> MY.NET.1.2:53
TCP TTL:23 TOS:0x0 ID:39426
SF** Seq: 0x50AE258C Ack: 0x39927E7D Win: 0x404
00 00 00 00 00 00 ……

Note that Tcpdump shows the SF (SYN and FIN) flags as being set.

Repeat Offenders

It should be noted than neither of the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers.

Summary

As these are invalid packets often used for reconnaissance purposes, it is recommended that such packets be blocked at the external router or firewall[19].

4.4 Watchlist 000222 IL-ISDNNET-990517

Alert Overview

This alert is triggered as a result of the IDS sensing the Watchlist 000222 IL-ISDNNET-990517 signature. Further information on this attack is not currently available.

Trend

The following table shows the trend in alerts for this attack over the two-month period.

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
732	123	385	1152

There is a general increase in these alerts.

Sources Identified

The following sources were identified as the source of these alerts:

159.226.5.22	159.226.91.20
159.226.47.14	159.226.92.10
159.226.47.196	159.226.111.1
159.226.47.217	159.226.115.1
159.226.61.62	159.226.120.9
159.226.63.200	159.226.228.1
159.226.66.130	

This range is based in China: 159.226.0.0: The Computer Network Center Chinese Academy of Sciences (NET-NCFC). Beijing 100080, China

Repeat Offenders

It should be noted than none of the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers.

Summary

If there is no business need to communicate with these systems in China then access from the 159.226.x.x networks above should be blocked at the external router (or

[19] Depending on the capabilities of the perimeter defence.

firewall).

If there is a need to communicate with systems in China it is recommended that such communication be restricted to designated hosts where possible. Extra monitoring is recommended. Target systems should be checked for vulnerabilities. Log analysis frequency should be increased.

4.5 Wingate 1080 Attempt

Alert Overview & Definition

This is attempted connection to port 1080. Whilst the Snort rule set in use at GIAC reports this as Wingate 1080, Whitehats refers to this as a scan for SOCKS servers.

http://www.whitehats.com IDS175 : Someone is scanning your system to see if it is running SOCKS. This may be a hacker that desires to "bounce" traffic through your system at other people. It may also be a chat server trying to determine if someone is indeed bouncing through your system to chat anonymously.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert tcp $EXTERNAL_NET any -> $HOME_NET 1080 (msg: "Wingate 1080 attempt"; ack: 0; flags: S;)

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
457	326	736	643

The incidence of this alert is reasonably consistent across the time period analysed.

Sources Identified

120 different hosts were identified as attempting this attack.

Repeat Offenders

It should be noted than only two of the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers. These are shown below:

24.4.196.167	cc32281-a.etntwn1.nj.home.com	Also involved in 515 from outside alert
61.139.110.69	<no DNS name>	Also involved in connect to SUN RPC high port alert

Summary

If the SOCKS or Wingate products are not in use, then the Firewall should be directed to block externally sourced sessions that attempt to use port 1080. This will require a stateful Firewall since this ephemeral port might also be used in a benign manner.

4.6 Sun RPC High Port Access

Alert Overview & Definition

This alert is triggered as a result of the IDS sensing an attempt to connect to the Sun RPC high ports. The port targeted was 32771. This port is associated with portmapper.

http://www.whitehats.com IDS429 : A query was sent to the rpcbind/portmap daemon on a solaris machine, requesting port information for rpc services.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert tcp $EXTERNAL_NET any -> $HOME_NET 32771 (msg: " Sun RPC High Port Access "; flags: A+; rpc:100000,*,*;)

This alert is candidate CAN-1999-0632 for inclusion in the CVE

Trend

The following table shows the trend in alerts for this signature over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
352	78	23	31

There is a significant reduction in the incidence of this alert.

Sources Identified

The following sources were identified as the source of these alerts:

205.188.153.99 thru 205.188.153.111

The source addresses are consecutive. It is possible they are spoofed. The netblock is assigned to: America Online, Inc (NETBLK-AOL-DTC), 22080 Pacific Blvd, Sterling, VA 20166, US.

Other sources are given below:

24.189.31.228 ool-18bd1fe4.dyn.optonline.net

64.4.13.74	msgr-sb5.msgr.hotmail.com
128.169.50.34	HELIOS.TNS.UTK.EDU
205.188.4.6	<no DNS name available>
205.188.7.102	<no DNS name available>
216.35.221.79	<no DNS name available>

Repeat Offenders

It should be noted than none of the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers.

Destination Systems

It should be noted that a significant proportion of the RPC connect attempts were targeted at the following two hosts:

MY.NET.224.138 MY.NET.213.159

Summary

The above two internal systems should be checked for vulnerabilities and signs of compromise.

4.7 Attempted Sun RPC High Port Access

Alert Overview

This alert is triggered as a result of the IDS sensing an attempt to connect to the Sun RPC high ports. This alert is akin to the one shown in the previous section.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
352	873	307	208

The incidence of this alert is reasonably consistent across the time period analysed.

Sources Identified

The following sources triggered this alert.

24.180.202.45	<no DNS name available>
152.163.241.59	<no DNS name available>
205.188.153.139	<no DNS name available>
209.10.41.242	<no DNS name available>

216.10.12.2	<no DNS name available>
216.10.12.30	<no DNS name available>
216.34.243.246	<no DNS name available>
216.99.200.242	securedesign.net
216.148.218.160	<no DNS name available>

It should also be noted that a consecutive set of source addresses were used: 205.188.153.100 thru 109 (omitting a couple of addresses). This address range resolves to icq.aol.com.

Destination Systems

Two specific hosts are targeted in the majority of these alerts. They are:

MY.NET.99.51 MY.NET.213.158

Repeat Offenders

The above systems do not appear in the logs for any other alert.

Summary

Further research is required to determine if the above two internal systems are running RPC based services. If so they should be checked for signs of compromise. The Firewall rules may need to be tightened.

4.8 Queso Fingerprinting

Alert Overview & Definition

This alert is triggered as a result of the IDS sensing an attempt to fingerprint the Operating System of hosts using Queso.

http://www.whitehats.com: IDS 29: A remote user has used the Queso tool to determine the OS fingerprint of the server.

It is possible for this to be a false positive if RFC2461 Quality of Service techniques are in use in the network.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "SCAN Queso Fingerprint attempt"; ttl: >225; flags: S12; reference:arachnids,29;)

This alert is candidate CAN-1999-0454 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
227	24	126	333

The incidence of this alert is inconsistent across the time period analysed.

Sources Identified

The following sources were responsible for over 50% of all Queso alerts.

63.78.39.192	192.dsl7839.rcsis.com
134.2.214.47	pool4047.studentenheim.uni-tuebingen.de
206.65.191.229	<no DNS name available>

Repeat Offenders

The above source systems do not appear in the logs for any other alert.

Summary

Consideration should be given to blocking these source addresses at the external router (or firewall).

4.9 Broadcast Ping to Subnet 70

Alert Overview

This is not a standard Snort alert. Windows systems do not respond to pings to broadcast addresses but other operating systems do. As such it is assumed that the rule was added to help protect a system or network against some uncommon attack. Further information is requested from GIAC to support this.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
136	0	12	5

There is a significant reduction in the incidence of this alert.

Destination Systems

All the alerts pertain to the .70 subnet.

Summary

It is suggested that this traffic be blocked at the perimeter.

4.10 SNMP public access

These alerts were triggered by internal systems only, with two exceptions. The Snort rules should be tuned to prevent this alert where the source is an internal system.

External Sources Identified

128.46.156.231	ece156-dhcp-2.ecn.purdue.edu
128.183.38.30	cesdis6.gsfc.nasa.gov

Summary

SNMP information can be useful for reconnaissance purposes. It is suggested that this traffic be blocked at the perimeter.

4.11 Null Scan!

Alert Overview & Definition

This alert is triggered as a result of the IDS sensing packets that appear to be Null Scans.

http://www.whitehats.com IDS4 : 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. An attacker may be scanning your system by sending these specially formatted frames to see what services are available.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"SCAN NULL";flags:0; seq:0; ack:0; reference:arachnids,4;)

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
47	60	452	266

The incidence of this alert is inconsistent across the time period analysed.

Summary

These are common false positives. No further analysis is provided on these alerts.

4.12 Back Orifice

Alert Overview & Definition

This alert is triggered as a result of the IDS sensing an attempt to connect to the ports used by the Back Orifice Trojan.

http://www.whitehats.com. The Whitehats Snort rules base has four different patterns for this alert. These are IDS189, IDS397-400. They allow tracking of the individual Back Orifice commands in use.

This alert is generated as a result of a Snort rules similar to the one shown below:

alert udp $EXTERNAL_NET any -> $HOME_NET 31337 (msg:"BACKDOOR BackOrifice access"; content: "|ce63 d1d2 16e7 13cf 39a5 a586|"; reference:arachnids,399;)

This alert is candidate CAN-1999-0660 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
36	0	4	5

Summary

This is just 'Trojan trolling'. Such trolling is common on Internet connections. Since we do not see any internal system respond, no further analysis is necessary. There may be value in refining the signature to alert on internal sources only.

4.13 Tiny Fragments

Alert Overview

This alert is triggered as a result of the IDS sensing fragments that are smaller than is normal.

This alert is generated as a result of one of the following Snort directives:

preprocessor minifrag: 128
preprocessor defrag

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
25	125	3118	2068

There is a significant increase in the incidence of this alert.

Summary

These are usually common false positives. This can also be an attempt to avoid an IDS. No further analysis is provided on these alerts.

4.14 NMAP ping

Alert Overview

This alert is triggered as a result of the IDS sensing a ping with the characteristics of NMAP[20] or hping2 (see http://www.eaglenet.org/antirez/hping2.html)

This alert is generated as a result of a Snort rules similar to the one shown below:

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP Nmap2.36BETA or HPING2 Echo ";itype:8;dsize:0; reference:arachnids,162;)

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
23	19	136	376

There is a significant increase in the incidence of this alert.

Summary

This is reconnaissance. Whilst reconnaissance is not itself usually damaging to a system or network, it is often the pre-cursor to more focussed attacks.

4.15 Connect to 515 from outside

Alert Overview

This is a recently developed attack that targets Linux systems running the line printer daemon (this is not installed by default on RedHat 6.2 or 7.0).

http://www.whitehats.com: IDS 456 & 457: A remote Format string vulnerability in use_syslog() function in LPRng 3.6.24 allows remote attackers to execute arbitrary commands.

This alert is generated as a result of a Snort rules similar to the one shown below:

[20] Nmap is a commonly used scanning tool. See http://www.insecure.org/nmap for more information.

alert tcp $EXTERNAL_NET any -> $HOME_NET 515 (msg:"EXPLOIT LPRng overflow"; flags: A+;)

This alert is candidate CAN-2000-0917 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
25	422	1	1

There is a single fortnight in which this alert was triggered extensively. It is possible that the attack resulted in compromised hosts. This trend is cause for concern. Further research is recommended.

Sources Identified

The following sources triggered this alert.

24.160.143.196	cs160143-196.satx.rr.com[21]
24.4.196.167	cc32281-a.etntwn1.nj.home.com[22]
62.46.70.175	L0054P15.dipool.highway.telekom.at
128.61.36.117	r36h117.res.gatech.edu
172.161.186.125	ACA1BA7D.ipt.aol.com
192.118.36.9	<no DNS name available>

Summary

This is a scan for systems that may be vulnerable to this attack (i.e. are listening on port 515). The firewall should be expected to drop these packets in normal circumstances. There were nearly 3000 systems targeted. Any systems susceptible to this attack should be patched and checked for signs of compromise.

4.16 Connect to 515 from inside

Alert Overview

This is a recently developed attack that is very similar to the previous one. It targets Linux systems running the line printer daemon (this is not installed by default on RedHat 6.2 or 7.0).

http://www.whitehats.com: IDS 456 & 457: A remote Format string vulnerability in use_syslog() function in LPRng 3.6.24 allows remote attackers to execute arbitrary commands.

This alert is generated as a result of a Snort rules similar to the one shown below:

[21] This system was also responsible for one of the Wingate 1080 attacks
[22] This system was also responsible for one of the Wingate 1080 attacks

alert tcp $HOME_NET any -> $EXTERNAL_NET 515 (msg:"EXPLOIT LPRng overflow"; flags: A+; content)

This alert is candidate CAN-2000-0917 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
5	12	3	139

There is a significant increase in the incidence of this alert.

Sources Identified

The following sources triggered this alert.

MY.NET.60.16
MY.NET.60.38
MY.NET.70.38
MY.NET.98.151
MY.NET.99.244
MY.NET163.17
MY.NET.179.78
MY.NET.219.122
MY.NET.219.194
MY.NET.253.12

Destinations Identified

The following destinations were involved in this alert.

64.23.4.67	chimay.skynetweb.com
128.8.3.106	bay6.umd.edu
129.155.192.99	<no DNS name available>
131.204.205.101	<no DNS name available>
148.243.214.7	na-148-243-214-7.na.avantel.net.mx
212.187.65.135	c65135.upc-c.chello.nl
216.181.129.185	<no DNS name available>

Summary

Investigate the above Internal systems for signs of compromise. Consider blocking port 515 at the Firewall.

As these alerts are generated as a result on internal systems in communication with external systems, it is recommended that these systems be investigated further. Whilst it is possible to use lpr (the printing protocol assigned to port 515) over the

Internet, this is not common.

4.17 SMB Name Wildcard

Alert Overview

This is a technique for mapping user and system names on Netbios platforms. It is commonly seen on internal networks with Windows clients. As such only external sources are considered here. Mapping of hidden[23] network drives shows the same signature and would indicate a compromised host.

http://www.whitehats.com: IDS 334 - 340: A remote user is likely attempting to open a named pie using the IPC$ share (or other $ shares).

This alert is generated as a result of a Snort rules similar to the one shown below:

alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS SMB IPC$access";flags: A+;)

This alert is candidate CAN-1999-0621 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
1	25	100	370

There is a significant increase in the incidence of this alert.

Sources Identified

There were 60 sources for this alert.

Summary

The firewall should be expected to drop these packets in normal circumstances. If the perimeter defences are not configured to block Netbios traffic from untrusted hosts then there is significant potential for widespread compromise.

4.18 External RPC call

Alert Overview

http://www.whitehats.com: IDS 428: A query was sent to the portmap daemon,

[23] Designated with a $ appended to the share name.

requesting port information for rpc services

This alert is generated as a result of a Snort rule similar to the one shown below:

Alert tcp $EXTERNAL_NET any -> $HOME_NET 111 (msg:"External RPC call"; flags: A+; rpc: 100000,*,*;)

This alert is candidate CAN-1999-0632 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
3	24	20	12

The incidence of this alert is inconsistent across the time period analysed.

Sources Identified

The following sources triggered this alert.

61.9.26.50	<no DNS name available>
63.11.25.117	1Cust117.tnt1.yakima.wa.da.uu.net
130.212.20.72	rsensing2.sfsu.edu
148.228.125.215	<no DNS name available>
192.71.148.152	birx22ms1.teliamobile.net
195.57.62.153	<no DNS name available>
195.116.66.14	jsmala.polmoslancut.com.pl
202.84.134.141	<no DNS name available>
206.210.80.6	<no DNS name available>
208.185.235.100	sdsl-208-185-235-100.dsl.sjc.megapath.net
209.178.23.187	CBL187.pool010.CH001-riverside.dhcp.hs.earthlink.net
211.48.210.193	<no DNS name available>
211.50.30.241	<no DNS name available>

Summary

GIAC should consider dropping RPC traffic at the firewall.

4.19 Probable NMAP fingerprint attempt

Alert Overview

This alert is triggered as a result of the IDS sensing a fingerprinting attempt with the characteristics of NMAP[24].

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
2	0	4	2

The incidence of this alert is generally consistent across the time period analysed.

Sources Identified

The following sources triggered this alert.

24.113.198.51	cr859517-a.surrey1.bc.wave.home.com
130.239.129.109	fysgr456.sn.umu.se
153.19.144.207	<no DNS name available>
206.205.246.2	ns.isrd.net[25]
211.109.37.120	<no DNS name available>

Summary

This is reconnaissance or a false positive.

4.20 site exec - Possible wu-ftpd exploit - GIAC000623

Alert Overview

This alert is triggered as a result of the IDS sensing a possible wu-ftpd exploit.

http://www.whitehats.com: IDS 285 - 288: This signature represents a remote ftpd attack against wu-2.6.0. This probe is common in both the linux and bsd versions of the published exploit.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert TCP $EXTERNAL any -> $INTERNAL 21 (msg: "IDS288/ftp-wuftp260-venglin-bsd"; flags: A+; content: "|31c0 50 50 50 b07e cd80 31db 31c0|"; depth: 32;)

This alert is candidate CAN-2000-0574 for inclusion in the CVE.

Trend

The following table shows the trend in alerts for this attack over the two-month period

[24] Nmap is a commonly used scanning tool. See http://www.insecure.org/nmap more information.
[25] This system appears to be a nameserver. It also features in a number of other alerts. It is probably benign. Further research is recommended.

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
1	2	0	0

The incidence of this alert is minimal across the time period analysed.

Sources Identified

The following source triggered this alert.

24.23.255.246 cm47580-a.ftwrth1.tx.home.com

Destination Systems

One specific host is targeted:

MY.NET.130.98

Further research is required to determine if this system runs wu-ftpd. If it does then this shows active targeting. The server should be checked for signs of compromise if the wu-ftpd version is vulnerable.

Summary

Further research is required. Incoming FTP should be restricted at the Firewall.

4.21 Russian Dynamo – SANS Flash 28-jul-00

Alert Overview

No further information is available on this alert at this time.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
0	546	0	0

There is a single fortnight in which this alert was triggered extensively. It is possible that the attack resulted in compromised hosts. This trend is cause for concern. Further research is recommended.

Sources Identified

The following two sources triggered this alert.

MY.NET.205.138
194.87.6.38 38.6.87.194.dynamic.dol.ru

The external address range has been previously associated with this alert.

Summary

Investigate the above Internal system for signs of compromise.

4.22 Watchlist 000220 NET-NCFC

Alert Overview

This alert is triggered as a result of the IDS sensing the Watchlist 000220 NET-NCFC attack.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
0	123	385	31164

There is a substantial increase in the incidence of this alert.

Sources Identified

The identified sources were all 159.226.x.x.

This range is based in China: 159.226.0.0: The Computer Network Center Chinese Academy of Sciences (NET-NCFC). Beijing 100080, China.

If there is no business need to communicate with these systems in China then access from the 159.226.x.x networks above should be blocked at the external router (or firewall).

If there is a need to communicate with systems in China it is recommended that such communication be restricted to designated hosts where possible. Extra monitoring is recommended. Target systems should be checked for vulnerabilities. Log analysis frequency should be increased.

Repeat Offenders

It should be noted than none on the systems identified as sources for this alert appear as the source for any other alert during the time period the data covers.

4.23 SMTP Source Port Traffic

Alert Overview

This alert is triggered as a result of the IDS sensing the SMTP traffic.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
0	3	93	0

There is a single fortnight in which this alert was triggered. It is possible that the attack resulted in compromised hosts. This trend is cause for concern. Further research is recommended.

Sources Identified

The following sources triggered this alert.

63.11.25.117	1Cust117.tnt1.yakima.wa.da.uu.net
64.161.240.254	adsl-64-161-240-254.dsl.lsan03.pacbell.net
165.112.79.25	vismed.nida.nih.gov
206.132.27.156	irc.east.gblx.net
213.74.161.214	eu214.st161-net74.ip.superonlinecorporate.com

Destinations Identified

The approximately 100 internal systems were involved in this alert.

Repeat Offenders

It should be noted that the first second address shown above (adsl-64-161-240-254.dsl.lsan03.pacbell.net) was also involved in numerous SYN-FIN scans. It is possible that these scans allowed the attack to focus on a system that was vulnerable. It should also be noted that this address appears to be in ADSL space and is likely to change dynamically over time. A dynamic address is not usually appropriate for an SMTP server.

Summary

The internal servers should be checked for signs of compromise. The list of 100 is omitted from this report for clarity and will be provided separately.

4.24 Happy 99 Virus

Alert Overview

This alert is triggered as a result of the IDS sensing a known virus pattern.

Trend

The following table shows the trend in alerts for this attack over the two-month period.

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4

0	0	1	0

Sources Identified

The following source triggered this alert.

63.216.198.158 ffml.fanfic.com

Destinations Identified

The following internal system was the recipient of the virus.

MY.NET.6.47

Summary

The above internal system should be checked for signs of compromise. If there is no effective anti-virus software on this system then there may be a series of compromised systems.

When executed, the infected program opens a window entitled "Happy New Year 1999 !!" and shows a fireworks display to disguise its installation. This worm sends itself to other users when the infected computer is online.

Further information on Happy99 can be found at http://www.symantec.com/avcenter/venc/data/happy99.worm.html.

4.25 STATDX UDP Attack

Alert Overview

This alert is triggered as a result of the IDS sensing a possible attack against statd.

http://www.whitehats.com: IDS 442: A remote attacker may be attempting to exploit a vulnerable rpc.statd service using the statd linux exploit.

This alert is generated as a result of a Snort rule similar to the one shown below:

alert TCP $EXTERNAL any -> $INTERNAL any (msg: "IDS442/rpc-statdx-exploit"; flags: A+; content: "/bin|c74604|/sh";)

This alert is CVE-2000-0666.

Trend

The following table shows the trend in alerts for this attack over the two-month period

Fortnight 1	Fortnight 2	Fortnight 3	Fortnight 4
0	0	0	1

Sources Identified

The following source triggered this alert.

206.210.80.6

Destination Systems

One specific host is targeted:

MY.NET.6.15

Summary

Further research is required to determine if this system is running rpc.statd. If it is then this shows active targeting and the system should be inspected for signs of compromise.

5 Port Scan Assessment – Entire Period

5.1 Initial description

Port scans of systems and networks connected to untrusted networks are very common. Individually a port scan is difficult to protect against, though it is easier to alert on. Some firewalls have the ability to automatically block the sources of port scans. This can also present problems with self-imposed Denial of Service problems.

The Snort configuration can be modified to reduce the number of alerts on port scans. This tuning process should be considered for GIAC's Snort systems.

5.2 Total number of Scans

This section describes the Portscan activity seen in the reporting period.

There were a total of 38,156 port scan alerts triggered in the period covered. The following table shows the distribution:

Source Location	Total Entries	Total Number of Source Systems
External	7557	1976
Internal	30599	644

5.3 Internal vs. External

The port scan logs show both external and internal systems as the sources of the port scans. It is common for internal systems to trigger the port scan pre-processor in IDS systems. These are usually false positives. Internal Windows based systems will commonly cause false positive port scan alerts, as can traceroute commands etc.

Nevertheless a number of the scan alerts do indeed relate to internal systems that are generating malformed TCP packets for scanning purposes.

If this scanning activity is permitted by GIAC then the noise level in the scan logs can be reduced. A common solution to this issue is to exclude the internal networks from the Port Scan pre-processor. In Snort this can be achieved with the following directive

preprocessor portscan-ignorehosts:

If internal systems are to be included in the port scan system, then it is strongly recommended that the alert threshold be raised to mitigate the number of false positives.

5.4 Internally Sourced Postscan Hosts

The volume of internal port scans is high enough to significantly increase the complexity of providing effective assessment. As a result, the following systems have been singled out as worthy of further attention.

Each of these systems has generated more than 150 port scan entries. It should be noted that 150 is a somewhat arbitrary figure[26]. Following further investigation of the hosts listed below it may prove advisable to investigate all the internal systems that are listed as port-scan sources.

Portscan by Frequency of Appearance in Logs

Internal Source	Number of Appearances in PortScan log	Internal Source	Number of Appearances in PortScan log
MY.NET.217.150	6290	MY.NET.70.38	255
MY.NET.217.158	4935	MY.NET.6.35	202
MY.NET.100.230	3009	MY.NET.6.45	196
MY.NET.219.126	2203	MY.NET.6.47	183
MY.NET.253.24	2002	MY.NET.6.34	165
MY.NET.217.126	1492	MY.NET.186.16	165
MY.NET.217.182	1328	MY.NET.213.186	160
MY.NET.142.21	498	MY.NET.156.110	156
MY.NET.1.3	330	MY.NET.202.94	155
MY.NET.1.5	301	MY.NET.60.43	151
MY.NET.1.4	260	MY.NET.186.17	150

5.5 Port Scans – Examples

5.5.1 Malformed Packets

Stephen Northcutt states "Attackers use out-of-spec packets to perform network mapping and to evade some intrusion detection systems and firewalls"[27]. The following section discusses some of the malformed packets seen in the data received from GIAC.

5.5.2 Example Xmas Tree Packet

The following Tcpdump trace shows an Xmas Tree scan packet. The Xmas Tree packet has all the TCP flags set (Reserved 2 bit, SYN, FIN, RST, PSH, ACK, URG). This is not a valid TCP packet. This packet is being generated by an internal system (MY.NET.217.150). This is a stimulus packet (not a response). Someone (or

[26] Further information on internal systems that have generated fewer scan alerts is available on request.
[27] Northcutt Stephen et al. Intrusion Signatures and Analysis. New Riders Publishing 2000. 343

something) is producing these packets targeted at external systems.

01/11-00:04:01.932648 MY.NET.217.150:2340 -> 24.130.58.80:1815
TCP TTL:126 TOS:0x0 ID:1600 DF
21SFRPAU Seq: 0x27851EB Ack: 0x41BC46D4 Win: 0x5018
09 24 07 17 02 78 51 EB 41 BC 46 D4 00 FF 50 18 .$...xQ.A.F...P.
FD F8 1A EC 00 00 73 45 1D 29 0C 44 60 E8 83 DB ......sE.).D`...
7A 0A

5.5.3 Example SYN-FIN-URG Packet

The following Tcpdump trace shows a SYN-FIN-URG packet. The Xmas Tree packet has all the TCP flags set (Reserved 2 bit, SYN, FIN, URG). This is not a valid TCP packet. This packet is being generated by an internal system (MY.NET.217.150). This is a stimulus packet (not a response). Someone (or something) is producing these packets targeted at external systems.

Note that the same source system is shown here, on a different date.

01/16-18:16:57.792719 MY.NET.217.150:10 -> 24.130.58.80:2340
TCP TTL:126 TOS:0x0 ID:64639 DF
21SF*U Seq: 0x4CF0309 Ack: 0x41A394ED Win: 0x5010
TCP Options => EOL EOL
55 B2 48 1B 00 DB 79 96 44 85 U.H...y.D.

5.6 Port Scans – External Distribution

5.6.1 Port Scans – Major Single Hosts

Externally sourced port scans present more of an issue for the organisation.

The following systems have been logged as those generating the highest frequency of port scan alerts in the reporting period.

Where possible the DNS names associated with these hosts is provided.

24.3.0.36	pD900E12C.dip.t-dialin.net
24.7.86.215	pD950AF12.dip.t-dialin.net
24.113.198.51	pD901B88B.dip.t-dialin.net[28]
24.189.31.228	reston-gnap-ip-216012-229.dynamic.ziplink.net[29]
62.227.243.120	<no DNS name available>
63.78.39.192	securedesign.net[30]

[28] This host was also involved in 39 other attacks, including Null Scans and NMAP fingerprints against two specific internal hosts : MY.NET.105.120 and MY.NET.217.146.
[29] This host was also involved in 15 other attacks, including SMB Name Wildcard and SUNRPC high port attacks against MY.NET.217.150.
[30] This host was also involved in 432 other attacks via Queso fingerprinting.

152.163.206.134	pD900B996.dip.t-dialin.net
164.67.22.71	q3.quakeshit.com / ts12-62.dialup.bol.ucla.edu[31]
207.172.3.10	<no DNS name available>
212.64.74.169	pD950B6B6.dip.t-dialin.net
216.99.200.242	pD9010D74.dip.t-dialin.net

GIAC should monitor port scans and consider how much information is being provided to external agencies as a result of these port scans.

5.1.2 Port Scans – Major Single Hosts

The following networks appeared with significant regularity in the port scan data provided. Further consideration should be given to determining how much access these networks need, especially the dial-up consumer networks.

Caution is recommended before large-scale blocking of ISP address ranges. Tighter rules on the firewall or external perimeter defences can often obviate the need for such blocking.

home.com	cgocable.net	ppp.tpnet.pl	t-diallin.net
videotron.net	splitrock.net	algx.net	

5.1.3 Block at Perimeter Router

Whilst not a complete solution to the problem of port scanning, it suggested that GIAC consider temporarily blocking the more determined port scanners at the external perimeter routers. These blocks might be temporary in nature. This action serves to lighten the Firewall and IDS load and hence the analyst's effort is reduced.

[31] Note that at the time of initial analysis (March 15, 2001) this first name was registered against this IP address. Given the name chosen, this was probably a DNS hijack. The second name given is that registered at March 30, 2001, this is more likely to be the correct name for this address.

6 Conclusion

6.1 Compromised Systems

There is substantial evidence of compromised systems within GIAC's network. It is recommended that the forensic analysis and remedial activities proposed in this report be implemented as a matter of urgency.

Suspected or confirmed compromised hosts should be removed from service as a matter of urgency.

6.2 Flawed Firewall Security

The firewall security ruleset appears to be flawed. It appears to allow more access than would be preferred. eeek! Best Practices dictate a default denial policy. This does not appear to be in place at GIAC.

6.3 IDS Strategy

The IDS strategy in use at GIAC is deficient. Data should be examined more frequently. Data dissemination should be controlled and encrypted. Complete data files should be retained at all times. Multiple sensors should be deployed. The data from these multiple sensors should be analysed separately and as a whole.

6.4 eeek! Security Consultants

eeek! would like to thank GIAC for this opportunity to provide IDS analysis work. We trust you find the results valuable and we look forward to further opportunities to work with you to resolve the issues identified in this report and to improve the security of GIAC's network.

Last Updated: July 24th, 2017

Upcoming Training

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
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
Baltimore Fall 2017 - SEC503: Intrusion Detection In-Depth	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
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