# Global Information Assurance Certification Paper

*** Northcutt, great work. Multiple sites adds to the challenge. Good solid process, good accuracy, good clarity. 91 ***

# 10 Detects for SANS GIAC Intrusion Analyst Certification


## Igor Gashinsky


## April 6

**Notes:**

These detects come from different organizations, and the architecture of each organization will be discussed next to the detects. Most traffic has been gathered using Shadow IDS. Due to the security policies of most of the organizations, both source and destination IP's have been sanitized.

## Organization A

## Architecture:

This organizations architecture uses an unprotected DMZ composed primarily of Solaris machines for DNS, mail and web servers, the rest of the organization is protected by a Raptor Firewall. A Shadow sensor has been recently deployed inside the DMZ, and these are the results of that deployment.

## Detect #1

```
04:24:43.882203 scanner1.com.38682 > X.X.X.1.imap2: S 328716360:328716360(0) win 2048
04:24:43.882203 scanner1.com.38682 > X.X.X.1.telnet: S 328716360:328716360(0) win 2048
04:24:43.882203 scanner1.com.38682 > X.X.X.1.domain: S 328716360:328716360(0) win 2048
04:24:43.902203 scanner1.com.38682 > X.X.X.1.ftp: S 328716360:328716360(0) win 2048
04:24:43.942203 scanner1.com.38682 > X.X.X.2.imap2: S 4181572569:4181572569(0) win 2048
04:24:43.942203 scanner1.com.38682 > X.X.X.2.telnet: S 4181572569:4181572569(0) win 2048
04:24:43.942203 scanner1.com.38682 > X.X.X.2.domain: S 4181572569:4181572569(0) win 2048
04:24:43.972203 scanner1.com.38682 > X.X.X.2.ftp: S 4181572569:4181572569(0) win 2048
…
04:28:41.002307 scanner1.com.38682 > X.X.X.254.imap2: S 2470241665:2470241665(0) win 2048
04:28:41.002307 scanner1.com.38682 > X.X.X.254.telnet: S 2470241665:2470241665(0) win 2048
04:28:41.002307 scanner1.com.38682 > X.X.X.254.domain: S 2470241665:2470241665(0) win 2048
04:28:41.002307 scanner1.com.38682 > X.X.X.254.ftp: S 2470241665:2470241665(0) win 2048
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | This is a fast SYN-Only scan of the entire subnet looking for ftp, telnet, dns and imap ports on every machine. Note that the source port remains the same during the entire sweep. |
| *Analysis* | This is definitely a fairly noisy scan using TCP Half-open technique. The attacker is probably relying on the fact that this technique won't show up in the host logs, and is oblivious about the Shadow Sensor. The fact that the source port remains the same, and the scan took only 4 minutes to sweep the subnet, indicate to me crafted packets, and an automated tool, probably Nmap. Because he is only scanning for specific ports demonstrates that he is either looking for specific services, or wanted to minimize the noise level of his scan. It is also pretty clear that he is looking for Unix boxes, and not Windows. Since most of the machines in the DMZ are Solaris 2.6 boxes, the attacker missed scanning for SunRPC, indicating a lack of familiarity with the network, and potentially, not-so up-to date vulnerability knowledge. This indicates a reconnaissance attempt on this network by somebody who does not have any insider knowledge and a possibly low to medium skill set. |
| *Threat* | This is a targeted scan, however the attacker knows nothing about the architecture of the network, so the threat right now is **LOW**. |

## Detect #2

```
04:30:20.892203 scanner1.com.61424 > X.X.X.5.2000: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.301: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.1349: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.22289: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.567: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.1538: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.1352: S 537300484:537300484(0) win 2048
04:30:20.892203 scanner1.com.61424 > X.X.X.5.734: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.656: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.56: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.3128: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.1414: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.509: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.2041: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.727: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.424: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.2001: S 537300484:537300484(0) win 2048
04:30:21.172203 scanner1.com.61424 > X.X.X.5.850: S 537300484:537300484(0) win 2048

04:37:39.793064 scanner2.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.792203 scanner2.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.793349 scanner3.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.792203 scanner3.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.793638 scanner1.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.792203 scanner1.com.61482 > X.X.X.5.718: S 3497501261:3497501261(0) win 1024
04:37:39.794507 scanner2.com.61482 > X.X.X.5.1441: S 3497501261:3497501261(0) win 1024
04:37:39.792203 scanner2.com.61482 > X.X.X.5.1441: S 3497501261:3497501261(0) win 1024
04:37:39.794797 scanner3.com.61482 > X.X.X.5.1441: S 3497501261:3497501261(0) win 1024
04:37:39.792203 scanner3.com.61482 > X.X.X.5.1441: S 3497501261:3497501261(0) win 1024
…
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | Two minutes previously this IP swept this subnet. |
| *Technique* | This is actually two scans, however they are closely related. The first scan looks like an Nmap TCP Half-open scan (SYN Only) of the X.X.X.5 machine for all ports. The random destination port pattern is indicative of Nmap. The second scan is looks like an Nmap decoy scan of the same host. Note the source ports for the first scan are the same, as are the sequence numbers. For the second scan the source ports and the sequence numbers are the same for ALL 3 HOSTS! This definitely is indicative of forged packets, and the speed with which they are coming, as well as coordination is indicative of an automated tool. |
| *Analysis* | My best analysis of the situation is that since the first scan ran for less then half a second, the attacker stops the scan almost the same time he hit enter. Probably realizing how noisy this scan is about to be. Then 7 minutes later, he launched an even noisier scan, but this time using decoy's in order to hide what his IP really was. Unfortunately for him, since I had the first scan logged, it was not to hard to see the real IP among the decoys. This, to me, is an indication to an amateur attacker who does not know his tools too well. This scan is targeted at the DNS server, and swept through every port of the machine, indicating attackers interest in the machine.<br><br>An interesting side note about this decoy scan: apparently, a "feature" of Nmap is when a decoy scan is initiated, all the decoys will use the same sequence numbers, as well as the same source ports in the scan. This should be a fairly easily distinguishable signature of an Nmap decoy scan, at least until it is fixed in the new version. |
| *Threat* | Since this is a DNS server, a critical infrastructure component, and this is a repeat visitor, the Threat factor is **MEDIUM**, even though the machine is fully patched. One never knows what new bug has been discovered, and not posted to Bugtraq, that could be used on the machines. |

## Detect #3

```
04:45:07.962203 scanner1.com.41805 > X.X.X.200.430: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.981: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.31: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.5715: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.93: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.1023: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.ssh: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.147: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.2005: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.1365: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.621: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.919: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.668: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.1083: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.5550: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.835: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.33: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.1408: FP 0:0(0) win 1024 urg 0
04:45:07.962203 scanner1.com.41805 > X.X.X.200.640: FP 0:0(0) win 1024 urg 0
…
04:45:08.472203 scanner1.com.41805 > X.X.X.200.710: FP 0:0(0) win 1024 urg 0
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | This is the third visit from this IP in less then 20 minutes! |
| *Technique* | This looks like a X-Mas Scan (FIN, PSH, URG flags set), in order to evade detection of the same IP, All the source ports are the same, Sequence numbers are set to 0, and the destination ports are random. This scan is VERY fast. |
| *Analysis* | Same guy, in less then 20 minutes. This time, a X-Mas scan against a Mail server. This sweep is VERY fast, designed to evade detection, and sweeps all the 65535 ports of the machine. In light of the past 2 scans, it looks like the attacker is showing interest in this network. Note that the source ports is the same, and the sequence numbers are all 0's, and Since this is an X-Mas scan, this indicates forged packets. Also, if he is using Nmap, which is my suspicion, he has root on the machine he is scanning from to generate these packets. |
| *Threat* | **MEDIUM**. Even though this machine only has 1 port opened – port 25, the fact that such a targeted scan is pointed at this network is sufficient to raise the threat level. At this point, in my opinion, a traceback attempt is warranted, however in this organization that has to be OK's by management. |

**POST SCAN NOTE**:

Given the fast, targeted scan against the network, we recommended to management to allow us to attempt a traceback. They permitted it, and we contacted the ISP where the packets seemed to originate from. This IP seemed to have originated from their ISDN dial-up pool, and to our surprise they were very cooperative after we send them the logs, and provided us with the phone number, and ell as the contact name of the offending account, as well as the phone number that call was placed from. Apparently, for billing purposes, and fraud protection, they use caller ID to identify and log every dial-up attempt, and know what customer has what IP at any given time, as well as then phone number he is connecting from. Being armed with this information, we proceeded to call the number, and when a woman in what seemed to be her late thirty's answered, things weren't making much sense. However, after explaining that we saw an attempt to probe our network, and assuring her that we are not going to prosecute, and just wanted to find out what is going on, she admitted that she had a thirteen year old son, who just a couple of weeks ago got a new computer. Mystery solved! We impressed upon her that such actions are highly disruptive towards an organization, and possibly illegal, and she promised us that this would be dealt with promptly. From the tone of her voice it seemed that our young "h4x0r" Tommy isn't going to be playing with his new computer for a while.

## Organization B
## Architecture:

This organization also uses an unprotected DMZ, which consists of 3 machines: 2 DNS servers, and a honeypot. The filters are tuned to detect ANY traffic to the honeypot, any traffic to the rest of the unused class C, and any TCP SYN/UDP to the DNS servers that is not 22/tcp (ssh) and 53/udp (dns queries). It has been tested by the Security Staff that the DNS server has no need for 53/tcp, since all the queries are quite small, and could easily fit in a udp datagram.

Detect #4

```
09:24:47.700000 scanner.com > A.A.A.255: icmp: echo request
09:24:47.700000 scanner.com > A.A.A.0: icmp: echo request
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | This is a "ping" scan of the network using the old BSD and the new broadcast addresses |
| *Analysis* | This is a ping scan, meant for reconnaissance purposes, to map out the network. Since Windows machines will not respond to the .0 address, and the Unix machines will, the attacker could find out the operating system of the target subnet with only 2 packets. Perhaps the most dangerous part of this detect, is the fact that the packet even made it past the routers, and the Networking Group has been "strongly advised" to log onto the border routers, and use the "no ip directed-broadcast" feature of the Cisco routers. |
| *Threat* | **MEDIUM**, this is a reconnaissance scan, will be low as soon as the routers are "fixed", since until then, the site could be a SMURF Amplifier. |

## Detect #5

```
04:29:37.820000 scanner.com > A.B.C.2: icmp: echo request
…
05:04:16.452203 scanner.com.51031 > A.B.C.2.1540: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.374: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.558: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.472: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.393: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.151: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.130: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.72: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.1005: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.22273: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.551: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.2108: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.908: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.440: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.305: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.1356: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.681: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.674: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.308: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.850: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.956: . 3686755777:3686755777(0) ack 0 win 2048
05:04:16.452203 scanner.com.51031 > A.B.C.2.414: . 3686755777:3686755777(0) ack 0 win 2048
…
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | A very fast ACK scan of the Honeypot machine |
| *Analysis* | This appears to be a pure ACK scan targeting a Honeypot. The lone ping in the beginning, and the speed of the scan indicate a script, quite possibly the new 2.30BETA17 version of NMAP (the first to offer ACK scanning). This scanning technique is designed to be stealthy, and would penetrate most non-state-aware firewall implementation. It is intended for mapping the firewall rule-bases, by waiting for either a RST, and ICMP Unreachable, or nothing to come back from the port, and based on that determine the rulebase (more information could be found at http://www.insecure.org/nmap/index.html#new) . This indicates that the attacker is using the latest tools, and since he is scanning the Honeypot, is not familiar with the network. |
| *Threat* | **MEDIUM**, the attacker is probing the defenses of the Honeypot, in preparation for an attack. |

**Detect #6**

```
18:00:30.160000 A.A.A.3.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:39.660000 A.A.A.3.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:40.950000 A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:44.120000 A.A.A.3.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:45.670000 A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:48.870000 A.A.A.3.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:49.940000 A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
18:00:54.060000 A.A.A.3.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]

18:04:24.630000 0:0:c:7:ac:2 1:0:5e:A:B:C ip 62: A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
        45c0 0030 0000 0000 0211 af94 c761 6105
        e000 0002 07c1 07c1 001c cdd8 0000 1005
        0f6e 0200 3139 396e 6574 0000 c761 6101
18:04:29.610000 0:0:c:7:ac:2 1:0:5e:A:B:C ip 62: A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
        45c0 0030 0000 0000 0211 af94 c761 6105
        e000 0002 07c1 07c1 001c cdd8 0000 1005
        0f6e 0200 3139 396e 6574 0000 c761 6101
18:04:34.350000 0:0:c:7:ac:2 1:0:5e:A:B:C ip 62: A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
        45c0 0030 0000 0000 0211 af94 c761 6105
        e000 0002 07c1 07c1 001c cdd8 0000 1005
        0f6e 0200 3139 396e 6574 0000 c761 6101
18:04:39.000000 0:0:c:7:ac:2 1:0:5e:A:B:C ip 62: A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
        45c0 0030 0000 0000 0211 af94 c761 6105
        e000 0002 07c1 07c1 001c cdd8 0000 1005
        0f6e 0200 3139 396e 6574 0000 c761 6101
18:04:43.320000 0:0:c:7:ac:2 1:0:5e:A:B:C ip 62: A.A.A.2.1985 > ALL-ROUTERS.MCAST.NET.1985: udp 20 [tos 0xc0]
        45c0 0030 0000 0000 0211 af94 c761 6105
        e000 0002 07c1 07c1 001c cdd8 0000 1005
        0f6e 0200 3139 396e 6574 0000 c761 6101
```

| | |
|---|---|
| *Active Targeting* | **NO** |
| *History* | **None previous** |
| *Technique* | UDP broadcasts from port 1985 to a router multicast address from the routers |
| *Analysis* | A.A.A.2 and A.A.A.3 are Cisco 7500 routers set up in an HSRP configuration. This traffic was first picked up when initially setting up Shadow, and playing around with filters. At first, since this was not traffic I was familiar with, I started looking at the traffic with full packet load and Ethernet addresses. As soon as I saw them, I pulled up the routers ARP table, to see who this was intended for. After a little investigation, it became evident that the MAC address "0:0:c:7:ac:2" is the Cisco default HSRP "virtual interface" address. This data would indicate that UDP multicast traffic between two HSRP'd routers on port 1985 is their "heartbeat". After calling Cisco, and asking for more information, my suspicions were confirmed. |
| *Threat* | **NONE** |

Detect #7

```
14:24:43.822603 scanner1.com.38682 > A.A.A.1. netbios-ns: S 398716360:328716360(0) win 2048
14:25:43.880003 scanner1.com.38682 > A.A.A.2. netbios-ns: S 398716360:328716360(0) win 2048
14:26:44.383278 scanner1.com.38682 > A.A.A.3. netbios-ns: S 398716360:328716360(0) win 2048
14:27:43.800001 scanner1.com.38682 > A.A.A.4. netbios-ns: S 398716360:328716360(0) win 2048
…
18:37:43.882203 scanner1.com.38682 > A.A.A.254. netbios-ns: S 398716360:328716360(0) win 2048
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | TCP port 137 sweep of the entire subnet. Low and Slow with 1-minute intervals between hosts. Source port and sequence numbers are always the same |
| *Analysis* | The fact that the source ports and sequence numbers are always the same indicates manufactured packets. Due to precision of the scan (almost EXACTLY 60 seconds between hosts) it is most definitely an automated probe. The scanner is possibly doing reconnaissance of the Class C, looking for Windows machines. Given the timing (This scan was detected on April 4), it is a possible sign of the Chode/911 Virus probing the network. Luckily, there are no Windows machines allowed on that subnet. |
| *Threat* | **Low**, there are no Windows machines on that network. |

## @Home Cable Modem Network

These detects have been taken from my @home cable modem w/ Linux IPCHAINS

Detect #8

```
Apr  4 11:37:48 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=72 S=0x00
I=63499 F=0x0000 T=127 (#14)
Apr  4 11:37:53 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=89 S=0x00
I=63244 F=0x0000 T=127 (#14)
Apr  4 11:38:50 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=72 S=0x00
I=6926 F=0x0000 T=127 (#14)
Apr  4 11:38:55 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=89 S=0x00
I=7439 F=0x0000 T=127 (#14)
Apr  4 11:39:52 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=72 S=0x00
I=12560 F=0x0000 T=127 (#14)
Apr  4 11:39:57 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=89 S=0x00
I=12305 F=0x0000 T=127 (#14)
Apr  4 11:40:54 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=72 S=0x00
I=22546 F=0x0000 T=127 (#14)
Apr  4 11:40:59 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 some_one_else:1054 my_IP:161 L=89 S=0x00
I=23059 F=0x0000 T=127 (#14)
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | UDP query for port 161, source is always 1054, two packets per minute |
| *Analysis* | The source IP is another @home Cable Modem User on the same subnet. This, at first, appear to be a scan for machines running SNMP, since 161/UDP is the SNMP port. However, given the frequency of the packets, and after having sniffed the connection, this looked like a trial version of HP OpenView, trying to auto-discover the rest of the network, and broadcasting the "private" and "public" community strings. Since those strings are essentially SNMP passwords, I notified abuse@home.com, and asked them to contact this user and tell him to turn this "feature" off, before some one malicious decides to exploit it. |
| *Threat* | To me: **VERY LOW**; To Him: **VERY HIGH** |

## Detect #9

```
Mar 29 03:26:47 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 24.0.94.130:42138 my_IP:119 L=44 S=0x00
I=34155 F=0x0000 T=243 SYN (#9)
Mar 29 03:27:04 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 24.0.94.130:53950 my_IP:119 L=44 S=0x00
I=34156 F=0x0000 T=243 SYN (#9)
Mar 29 07:19:38 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 24.0.94.130:36132 my_IP:119 L=44 S=0x00
I=50615 F=0x0000 T=243 SYN (#9)
Mar 29 07:19:57 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 24.0.94.130:49313 my_IP:119 L=44 S=0x00
I=50616 F=0x0000 T=243 SYN (#9)
Mar 29 11:56:08 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 24.0.94.130:39787 my_IP:119 L=44 S=0x00
I=60637 F=0x0000 T=243 SYN (#9)

Nslookup 24.0.94.130
Server:  proxy1.union1.nj.home.com
Address:  X.X.X.33

Name:    authorized-scan.security.home.net
Address:  24.0.94.130
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | Similar scans have been showing up for the past 2 weeks |
| *Technique* | Scan to TCP port 119 (NNTP) at regular intervals of approx. 4 hours. Same IP |
| *Analysis* | This scan is directed at finding News Servers installed on the Cable Modem network ran by @Home. After contacting their customer support, I was informed that this is "for my benefit", since they have been experiencing slowdowns due to errant News Servers, and they are in violation of the Usage Agreement. |
| *Threat* | **VERY Low.** But, I do not appreciate the fact that "Big Brother is watching". |

## Detect #10

I apologize for the long detect, but this I found to be very interesting.

```
Apr  5 06:25:10 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 207.71.92.193:137 my_IP:137 L=78 S=0x00 I=42387 F=0x0000
T=115 (#15)
Apr  5 06:25:11 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 207.71.92.193:137 my_IP:137 L=78 S=0x00 I=30356 F=0x0000
T=115 (#15)
Apr  5 06:25:13 my_machine kernel: Packet log: input REJECT eth0 PROTO=17 207.71.92.193:137 my_IP:137 L=78 S=0x00 I=10389 F=0x0000
T=115 (#15)
Apr  5 06:25:30 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3376 my_IP:21 L=44 S=0x00 I=5279 F=0x4000 T=116
SYN (#10)
Apr  5 06:25:33 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3376 my_IP:21 L=44 S=0x00 I=28576 F=0x4000 T=116
SYN (#10)
Apr  5 06:25:39 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3376 my_IP:21 L=44 S=0x00 I=49059 F=0x4000 T=116
SYN (#10)
Apr  5 06:25:51 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3376 my_IP:21 L=44 S=0x00 I=22697 F=0x4000 T=116
SYN (#10)
Apr  5 06:26:15 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3472 my_IP:23 L=44 S=0x00 I=55734 F=0x4000 T=116
SYN (#10)
Apr  5 06:26:18 my_machine kernel: Packet log: input REJECT eth0PROTO=6 207.71.92.221:3472 my_IP:23 L=44 S=0x00 I=12728 F=0x4000 T=116
SYN (#10)
Apr  5 06:26:24 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3472 my_IP:23 L=44 S=0x00 I=35770 F=0x4000 T=116
SYN (#10)
Apr  5 06:26:36 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3472 my_IP:23 L=44 S=0x00 I=31424 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:00 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3533 my_IP:25 L=44 S=0x00 I=32203 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:03 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3533 my_IP:25 L=44 S=0x00 I=24524 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:09 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3533 my_IP:25 L=44 S=0x00 I=6094 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:21 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3533 my_IP:25 L=44 S=0x00 I=5843 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:45 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3609 my_IP:79 L=44 S=0x00 I=5085 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:48 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3609 my_IP:79 L=44 S=0x00 I=45278 F=0x4000 T=116
SYN (#10)
Apr  5 06:27:54 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3609 my_IP:79 L=44 S=0x00 I=55777 F=0x4000 T=116
SYN (#10)
Apr  5 06:28:06 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3609 my_IP:79 L=44 S=0x00 I=20454 F=0x4000 T=116
SYN (#10)
Apr  5 06:28:30 my_machine kernel: Packet log: input ACCEPT eth0 PROTO=6 207.71.92.221:3685 my_IP:80 L=44 S=0x00 I=1519 F=0x4000 T=116
SYN (#8)
Apr  5 06:28:30 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3686 my_IP:110 L=44 S=0x00 I=8175 F=0x4000 T=116
SYN (#10)
Apr  5 06:28:33 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3686 my_IP:110 L=44 S=0x00 I=24304 F=0x4000
T=116 SYN (#10)
Apr  5 06:28:39 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3686 my_IP:110 L=44 S=0x00 I=31475 F=0x4000
T=116 SYN (#10)
Apr  5 06:28:51 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3686 my_IP:110 L=44 S=0x00 I=8696 F=0x4000 T=116
SYN (#10)
Apr  5 06:29:15 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3747 my_IP:113 L=44 S=0x00 I=7427 F=0x4000 T=116
SYN (#10)
Apr  5 06:29:18 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3747 my_IP:113 L=44 S=0x00 I=9988 F=0x4000 T=116
SYN (#10)
Apr  5 06:29:24 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3747 my_IP:113 L=44 S=0x00 I=64261 F=0x4000
T=116 SYN (#10)
Apr  5 06:29:36 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3747 my_IP:113 L=44 S=0x00 I=56840 F=0x4000
T=116 SYN (#10)
Apr  5 06:30:01 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3801 my_IP:139 L=44 S=0x00 I=65295 F=0x4000
T=116 SYN (#10)
Apr  5 06:30:04 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3801 my_IP:139 L=44 S=0x00 I=31249 F=0x4000
T=116 SYN (#10)
Apr  5 06:30:10 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3801 my_IP:139 L=44 S=0x00 I=8467 F=0x4000 T=116
SYN (#10)
Apr  5 06:30:22 my_machine kernel: Packet log: input REJECT eth0 PROTO=6 207.71.92.221:3801 my_IP:139 L=44 S=0x00 I=38166 F=0x4000
T=116 SYN (#10)
```

| | |
|---|---|
| *Active Targeting* | **YES** |
| *History* | **None previous** |
| *Technique* | This appears to be a port scan of "well-known" ports. The source port varies, and it is always 4 packets per destination port. |
| *Analysis* | This port scan looked a bit anomalous, in comparison to other port scans that have been targeted at my machine, since it used 4 packets per port, and was fairly slow, so I decided to investigate. The IP address resolved to "shieldsup.grc.com", so I went and visited their web-site. After looking around, I saw a "Scan my Ports" and "Probe my Shields" buttons, so I clicked them both, and lo and behold, the same IP scanned me. However, this time it was a stimulus-response pair, since my clicking on those buttons stimulated a portscan. Since I am the only user on my cable modem, and never before visited this site, the fact that it scanned me was a little odd. In my mind, there are 2 possibilities, somebody spoofed my IP, and send a request using it to scan me, or Shield's Up is randomly scanning Planet Earth. I find possibility 2 disturbing, but possibility 1 even more so. If somebody could ask shield's up to scan somebody else's machine, and sniff out the results of the scan, they can effectively probe everyone's networks and can not be traced back whatsoever. Scary thought! |
| *Threat* | **POTENTIALLY HIGH** |

Last Updated: January 18th, 2018

# Upcoming Training


| | | | |
|---|---|---|---|
| Las Vegas 2018 - SEC503: Intrusion Detection In-Depth | Las Vegas, NV | Jan 28, 2018 - Feb 02, 2018 | vLive |
| SANS Las Vegas 2018 | Las Vegas, NV | Jan 28, 2018 - Feb 02, 2018 | Live Event |
| SANS London February 2018 | London, United Kingdom | Feb 05, 2018 - Feb 10, 2018 | Live Event |
| SANS Dallas 2018 | Dallas, TX | Feb 19, 2018 - Feb 24, 2018 | Live Event |
| Community SANS Baltimore SEC503 | Baltimore, MD | Mar 12, 2018 - Mar 17, 2018 | Community SANS |
| SANS Northern VA Spring - Tysons 2018 | McLean, VA | Mar 17, 2018 - Mar 24, 2018 | Live Event |
| SANS Secure Canberra 2018 | Canberra, Australia | Mar 19, 2018 - Mar 24, 2018 | Live Event |
| SANS 2018 | Orlando, FL | Apr 03, 2018 - Apr 10, 2018 | Live Event |
| SANS Abu Dhabi 2018 | Abu Dhabi, United Arab Emirates | Apr 07, 2018 - Apr 12, 2018 | Live Event |
| SANS London April 2018 | London, United Kingdom | Apr 16, 2018 - Apr 21, 2018 | Live Event |
| SANS Baltimore Spring 2018 | Baltimore, MD | Apr 21, 2018 - Apr 28, 2018 | Live Event |
| Baltimore Spring 2018 - SEC503: Intrusion Detection In-Depth | Baltimore, MD | Apr 23, 2018 - Apr 28, 2018 | vLive |
| SANS vLive - SEC503: Intrusion Detection In-Depth | SEC503 - 201805, | May 02, 2018 - Jun 14, 2018 | vLive |
| Community SANS Virginia Beach SEC503 | Virginia Beach, VA | May 07, 2018 - May 12, 2018 | Community SANS |
| SANS Security West 2018 | San Diego, CA | May 11, 2018 - May 18, 2018 | Live Event |
| SANS Oslo June 2018 | Oslo, Norway | Jun 18, 2018 - Jun 23, 2018 | Live Event |
| SANS Minneapolis 2018 | Minneapolis, MN | Jun 25, 2018 - Jun 30, 2018 | Live Event |
| SANSFIRE 2018 | Washington, DC | Jul 14, 2018 - Jul 21, 2018 | Live Event |
| Security Operations Summit & Training 2018 | New Orleans, LA | Jul 30, 2018 - Aug 06, 2018 | Live Event |
| SANS San Antonio 2018 | San Antonio, TX | Aug 06, 2018 - Aug 11, 2018 | Live Event |
| Community SANS Columbia SEC503 | Columbia, MD | Aug 13, 2018 - Aug 18, 2018 | Community SANS |
| SANS Virginia Beach 2018 | Virginia Beach, VA | Aug 20, 2018 - Aug 31, 2018 | Live Event |
| SANS OnDemand | Online | Anytime | Self Paced |
| SANS SelfStudy | Books & MP3s Only | Anytime | Self Paced |