Global Information Assurance Certification Paper

Sven Olensky

Practical Assignment
SANS Intrusion Detection Track 3
New Orleans 01/2001

Contents

0. Preface

This documentation is the result of my work on the assignments in which completion is required to obtain the certification of the GCIA, the GIAC Certified Intrusion Analyst. I worked the assignments that had the version 2.7a, last updated March 1st 2001 (see [GCIA]).

It consists of three major parts:

- Five network traces and their analysis
- Describe the State of Intrusion Detection
- "Analyze This"- Scenario

That being said, I hope you enjoy reading this!

Sven Olensky
April 4th 2001
Atlanta, GA.

1. Assignment 1: Network Detects

1.1. Introduction

You will find 5 different traces / attacks in this chapter, together with a complete analysis.

1.2. Incident 1

tcpdump output:

```
20:47:21.034593 193.252.111.151.1811 > AAA.BBB.CCC.3.www: S 4201115446:4201115446(0) win 16384 <mss 1452> (DF)
20:47:21.175221 193.252.111.151.radacct > AAA.BBB.CCC.5.www: S 4201414990:4201414990(0) win 16384 <mss 1452>
(DF)
20:47:23.444599 193.252.111.151.1834 > AAA.BBB.CCC.16.www: S 4204893175:4204893175(0) win 16384 <mss 1452> (DF)
20:47:23.748518 193.252.111.151.1840 > AAA.BBB.CCC.19.www: S 4205746136:4205746136(0) win 16384 <mss 1452> (DF)
20:47:23.754542 AAA.BBB.CCC.19.www > 193.252.111.151.1840: S 950796544:950796544(0) ack 4205746137 win 16384
<mss 1452> (DF)
20:47:23.916062 193.252.111.151.1840 > AAA.BBB.CCC.19.www: . ack 1 win 17424 (DF)
20:47:23.924765 193.252.111.151.1840 > AAA.BBB.CCC.19.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:24.169434 193.252.111.151.radacct > AAA.BBB.CCC.5.www: S 4201414990:4201414990(0) win 16384 <mss 1452>
(DF)
20:47:24.931787 193.252.111.151.1840 > AAA.BBB.CCC.19.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:25.448442 193.252.111.151.1852 > AAA.BBB.CCC.25.www: S 4207659234:4207659234(0) win 16384 <mss 1452> (DF)
20:47:25.448755 AAA.BBB.CCC.25.www > 193.252.111.151.1852: S 3701098786:3701098786(0) ack 4207659235 win 8192
<mss 1460>
20:47:25.609619 193.252.111.151.1852 > AAA.BBB.CCC.25.www: . ack 1 win 17520 (DF)
20:47:25.617365 193.252.111.151.1852 > AAA.BBB.CCC.25.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:25.618071 AAA.BBB.CCC.25.www > 193.252.111.151.1852: P 1:119(118) ack 19 win 8192
20:47:25.618351 AAA.BBB.CCC.25.www > 193.252.111.151.1852: F 119:119(0) ack 19 win 8192
20:47:25.779813 193.252.111.151.1852 > AAA.BBB.CCC.25.www: . ack 120 win 17520 (DF)
20:47:25.786265 193.252.111.151.1852 > AAA.BBB.CCC.25.www: F 19:19(0) ack 120 win 17520 (DF)
20:47:25.786507 AAA.BBB.CCC.25.www > 193.252.111.151.1852: . ack 20 win 8192
20:47:25.824941 193.252.111.151.1860 > AAA.BBB.CCC.25.www: S 4208796120:4208796120(0) win 16384 <mss 1452> (DF)
20:47:25.825250 AAA.BBB.CCC.25.www > 193.252.111.151.1860: S 3701248985:3701248985(0) ack 4208796121 win 8192
<mss 1460>
20:47:25.965812 193.252.111.151.1863 > AAA.BBB.CCC.30.www: S 4209190534:4209190534(0) win 16384 <mss 1452> (DF)
20:47:25.966188 AAA.BBB.CCC.30.www > 193.252.111.151.1863: S 84218977:84218977(0) ack 4209190535 win 8712 <mss
1460> (DF)
20:47:25.982814 193.252.111.151.1860 > AAA.BBB.CCC.25.www: . ack 1 win 17520 (DF)
20:47:25.992719 193.252.111.151.1860 > AAA.BBB.CCC.25.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:25.993425 AAA.BBB.CCC.25.www > 193.252.111.151.1860: P 1:119(118) ack 19 win 8192
20:47:25.993703 AAA.BBB.CCC.25.www > 193.252.111.151.1860: F 119:119(0) ack 19 win 8192
20:47:26.130133 193.252.111.151.1863 > AAA.BBB.CCC.30.www: . ack 1 win 17520 (DF)
20:47:26.138178 193.252.111.151.1863 > AAA.BBB.CCC.30.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:26.140197 AAA.BBB.CCC.30.www > 193.252.111.151.1863: P 1:144(143) ack 19 win 8694 (DF)
20:47:26.140377 AAA.BBB.CCC.30.www > 193.252.111.151.1863: F 144:144(0) ack 19 win 8694 (DF)
20:47:26.155127 193.252.111.151.1860 > AAA.BBB.CCC.25.www: . ack 120 win 17520 (DF)
20:47:26.163046 193.252.111.151.1860 > AAA.BBB.CCC.25.www: F 19:19(0) ack 120 win 17520 (DF)
20:47:26.163292 AAA.BBB.CCC.25.www > 193.252.111.151.1860: . ack 20 win 8192
20:47:26.302228 193.252.111.151.1863 > AAA.BBB.CCC.30.www: . ack 145 win 17520 (DF)
20:47:26.310874 193.252.111.151.1863 > AAA.BBB.CCC.30.www: F 19:19(0) ack 145 win 17520 (DF)
20:47:26.310960 AAA.BBB.CCC.30.www > 193.252.111.151.1863: . ack 20 win 8694 (DF)
20:47:26.421070 193.252.111.151.1871 > AAA.BBB.CCC.30.www: S 4210253060:4210253060(0) win 16384 <mss 1452> (DF)
20:47:26.421159 AAA.BBB.CCC.30.www > 193.252.111.151.1871: S 84218986:84218986(0) ack 4210253061 win 8712 <mss
1460> (DF)
20:47:26.444951 193.252.111.151.1834 > AAA.BBB.CCC.16.www: S 4204893175:4204893175(0) win 16384 <mss 1452> (DF)
20:47:26.584538 193.252.111.151.1871 > AAA.BBB.CCC.30.www: . ack 1 win 17520 (DF)
20:47:26.592166 193.252.111.151.1871 > AAA.BBB.CCC.30.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:26.592761 AAA.BBB.CCC.30.www > 193.252.111.151.1871: P 1:144(143) ack 19 win 8694 (DF)
20:47:26.592931 AAA.BBB.CCC.30.www > 193.252.111.151.1871: F 144:144(0) ack 19 win 8694 (DF)
20:47:26.756476 193.252.111.151.1871 > AAA.BBB.CCC.30.www: . ack 145 win 17520 (DF)
20:47:26.761467 193.252.111.151.1871 > AAA.BBB.CCC.30.www: F 19:19(0) ack 145 win 17520 (DF)
20:47:26.761542 AAA.BBB.CCC.30.www > 193.252.111.151.1871: . ack 20 win 8694 (DF)
20:47:26.911230 193.252.111.151.1840 > AAA.BBB.CCC.19.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:26.930358 193.252.111.151.1882 > AAA.BBB.CCC.39.www: S 4212028575:4212028575(0) win 16384 <mss 1452> (DF)
20:47:26.931078 AAA.BBB.CCC.39.www > 193.252.111.151.1882: S 1609731327:1609731327(0) ack 4212028576 win 8712
<mss 1460> (DF)
20:47:27.089292 193.252.111.151.1882 > AAA.BBB.CCC.39.www: . ack 1 win 17520 (DF)
```

```
20:47:27.097163 193.252.111.151.1882 > AAA.BBB.CCC.39.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:27.100954 193.252.111.151.1885 > AAA.BBB.CCC.30.www: S 4212350101:4212350101(0) win 16384 <mss 1452> (DF)
20:47:27.101035 AAA.BBB.CCC.30.www > 193.252.111.151.1885: S 84218992:84218992(0) ack 4212350102 win 8712 <mss
1460> (DF)
20:47:27.218032 AAA.BBB.CCC.39.www > 193.252.111.151.1882: P 1:144(143) ack 19 win 8694 (DF)
20:47:27.218298 AAA.BBB.CCC.39.www > 193.252.111.151.1882: F 144:144(0) ack 19 win 8694 (DF)
20:47:27.260053 193.252.111.151.1889 > AAA.BBB.CCC.42.www: S 4212962099:4212962099(0) win 16384 <mss 1452> (DF)
20:47:27.260497 AAA.BBB.CCC.42.www > 193.252.111.151.1889: S 207445:207445(0) ack 4212962100 win 8712 <mss 1460>
(DF)
20:47:27.266927 193.252.111.151.1885 > AAA.BBB.CCC.30.www: . ack 1 win 17520 (DF)
20:47:27.300750 193.252.111.151.1885 > AAA.BBB.CCC.30.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:27.302117 AAA.BBB.CCC.30.www > 193.252.111.151.1885: P 1:470(469) ack 66 win 8647 (DF)
20:47:27.302249 AAA.BBB.CCC.30.www > 193.252.111.151.1885: F 470:470(0) ack 66 win 8647 (DF)
20:47:27.379205 193.252.111.151.1882 > AAA.BBB.CCC.39.www: . ack 145 win 17520 (DF)
20:47:27.386644 193.252.111.151.1891 > AAA.BBB.CCC.43.www: S 4213441535:4213441535(0) win 16384 <mss 1452> (DF)
20:47:27.401769 193.252.111.151.1882 > AAA.BBB.CCC.39.www: F 19:19(0) ack 145 win 17520 (DF)
20:47:27.402224 AAA.BBB.CCC.39.www > 193.252.111.151.1882: . ack 20 win 8694 (DF)
20:47:27.426649 193.252.111.151.1889 > AAA.BBB.CCC.42.www: . ack 1 win 17520 (DF)
20:47:27.433136 193.252.111.151.1889 > AAA.BBB.CCC.42.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:27.475987 193.252.111.151.1885 > AAA.BBB.CCC.30.www: . ack 471 win 17520 (DF)
20:47:27.481815 193.252.111.151.1885 > AAA.BBB.CCC.30.www: F 66:66(0) ack 471 win 17520 (DF)
20:47:27.481900 AAA.BBB.CCC.30.www > 193.252.111.151.1885: . ack 67 win 8647 (DF)
20:47:27.488191 193.252.111.151.1893 > AAA.BBB.CCC.39.www: S 4213679486:4213679486(0) win 16384 <mss 1452> (DF)
20:47:27.488864 AAA.BBB.CCC.39.www > 193.252.111.151.1893: S 1609731345:1609731345(0) ack 4213679487 win 8712
<mss 1460> (DF)
20:47:27.605503 AAA.BBB.CCC.42.www > 193.252.111.151.1889: . ack 19 win 8694 (DF)
20:47:27.647303 193.252.111.151.1893 > AAA.BBB.CCC.39.www: . ack 1 win 17520 (DF)
20:47:27.655229 193.252.111.151.1893 > AAA.BBB.CCC.39.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:27.656272 AAA.BBB.CCC.39.www > 193.252.111.151.1893: P 1:144(143) ack 19 win 8694 (DF)
20:47:27.656423 AAA.BBB.CCC.39.www > 193.252.111.151.1893: F 144:144(0) ack 19 win 8694 (DF)
20:47:27.825200 193.252.111.151.1893 > AAA.BBB.CCC.39.www: . ack 145 win 17520 (DF)
20:47:27.831506 193.252.111.151.1893 > AAA.BBB.CCC.39.www: F 19:19(0) ack 145 win 17520 (DF)
20:47:27.831930 AAA.BBB.CCC.39.www > 193.252.111.151.1893: . ack 20 win 8694 (DF)
20:47:28.054329 193.252.111.151.1901 > AAA.BBB.CCC.39.www: S 4214606804:4214606804(0) win 16384 <mss 1452> (DF)
20:47:28.055031 AAA.BBB.CCC.39.www > 193.252.111.151.1901: S 1609731363:1609731363(0) ack 4214606805 win 8712
<mss 1460> (DF)
20:47:28.216749 193.252.111.151.1901 > AAA.BBB.CCC.39.www: . ack 1 win 17520 (DF)
20:47:28.227865 193.252.111.151.1901 > AAA.BBB.CCC.39.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:28.281295 AAA.BBB.CCC.39.www > 193.252.111.151.1901: P 1:891(890) ack 66 win 8647 (DF)
20:47:28.281299 AAA.BBB.CCC.39.www > 193.252.111.151.1901: F 891:891(0) ack 66 win 8647 (DF)
20:47:28.404881 AAA.BBB.CCC.42.www > 193.252.111.151.1889: P 1:200(199) ack 19 win 8694 (DF)
20:47:28.461598 193.252.111.151.1901 > AAA.BBB.CCC.39.www: . ack 892 win 17520 (DF)
20:47:28.469760 193.252.111.151.1901 > AAA.BBB.CCC.39.www: F 66:66(0) ack 892 win 17520 (DF)
20:47:28.470157 AAA.BBB.CCC.39.www > 193.252.111.151.1901: . ack 67 win 8647 (DF)
20:47:28.493145 AAA.BBB.CCC.42.www > 193.252.111.151.1889: F 200:200(0) ack 19 win 8694 (DF)
20:47:28.651604 193.252.111.151.1889 > AAA.BBB.CCC.42.www: . ack 201 win 17520 (DF)
20:47:28.656298 193.252.111.151.1889 > AAA.BBB.CCC.42.www: F 19:19(0) ack 201 win 17520 (DF)
20:47:28.656432 AAA.BBB.CCC.42.www > 193.252.111.151.1889: . ack 20 win 8694 (DF)
20:47:28.662493 193.252.111.151.1902 > AAA.BBB.CCC.42.www: S 4214879593:4214879593(0) win 16384 <mss 1452> (DF)
20:47:28.662652 AAA.BBB.CCC.42.www > 193.252.111.151.1902: S 207447:207447(0) ack 4214879594 win 8712 <mss 1460>
(DF)
20:47:28.820770 193.252.111.151.1902 > AAA.BBB.CCC.42.www: . ack 1 win 17520 (DF)
20:47:28.828993 193.252.111.151.1902 > AAA.BBB.CCC.42.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:28.831288 AAA.BBB.CCC.42.www > 193.252.111.151.1902: P 1:200(199) ack 19 win 8694 (DF)
20:47:28.831900 AAA.BBB.CCC.42.www > 193.252.111.151.1902: F 200:200(0) ack 19 win 8694 (DF)
20:47:28.995963 193.252.111.151.1902 > AAA.BBB.CCC.42.www: . ack 201 win 17520 (DF)
20:47:29.001248 193.252.111.151.1902 > AAA.BBB.CCC.42.www: F 19:19(0) ack 201 win 17520 (DF)
20:47:29.001377 AAA.BBB.CCC.42.www > 193.252.111.151.1902: . ack 20 win 8694 (DF)
20:47:29.343912 193.252.111.151.1910 > AAA.BBB.CCC.42.www: S 4216227020:4216227020(0) win 16384 <mss 1452> (DF)
20:47:29.344065 AAA.BBB.CCC.42.www > 193.252.111.151.1910: S 207463:207463(0) ack 4216227021 win 8712 <mss 1460>
(DF)
20:47:29.467303 193.252.111.151.1913 > AAA.BBB.CCC.51.www: S 4216569590:4216569590(0) win 16384 <mss 1452> (DF)
20:47:29.467558 AAA.BBB.CCC.51.www > 193.252.111.151.1913: S 1824861410:1824861410(0) ack 4216569591 win 16384
<mss 512>
20:47:29.505696 193.252.111.151.1910 > AAA.BBB.CCC.42.www: . ack 1 win 17520 (DF)
20:47:29.515614 193.252.111.151.1910 > AAA.BBB.CCC.42.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:29.537089 AAA.BBB.CCC.42.www > 193.252.111.151.1910: P 1:605(604) ack 66 win 8647 (DF)
20:47:29.537360 AAA.BBB.CCC.42.www > 193.252.111.151.1910: F 605:605(0) ack 66 win 8647 (DF)
20:47:29.628599 193.252.111.151.1913 > AAA.BBB.CCC.51.www: . ack 1 win 16384 (DF)
20:47:29.635985 193.252.111.151.1913 > AAA.BBB.CCC.51.www: P 1:19(18) ack 1 win 16384 (DF)
20:47:29.636498 AAA.BBB.CCC.51.www > 193.252.111.151.1913: P 1:166(165) ack 19 win 16384
20:47:29.636501 AAA.BBB.CCC.51.www > 193.252.111.151.1913: F 166:166(0) ack 19 win 16384
20:47:29.709092 193.252.111.151.1910 > AAA.BBB.CCC.42.www: . ack 606 win 17520 (DF)
20:47:29.749811 193.252.111.151.1910 > AAA.BBB.CCC.42.www: F 66:66(0) ack 606 win 17520 (DF)
20:47:29.749940 AAA.BBB.CCC.42.www > 193.252.111.151.1910: . ack 67 win 8647 (DF)
20:47:29.805779 193.252.111.151.1913 > AAA.BBB.CCC.51.www: . ack 167 win 16384 (DF)
20:47:29.815801 193.252.111.151.1913 > AAA.BBB.CCC.51.www: F 19:19(0) ack 167 win 16384 (DF)
20:47:29.815883 AAA.BBB.CCC.51.www > 193.252.111.151.1913: . ack 20 win 16384
20:47:29.851147 193.252.111.151.1920 > AAA.BBB.CCC.51.www: S 4217599665:4217599665(0) win 16384 <mss 1452> (DF)
20:47:29.851393 AAA.BBB.CCC.51.www > 193.252.111.151.1920: S 1824914048:1824914048(0) ack 4217599666 win 16384
<mss 512>
20:47:30.010772 193.252.111.151.1920 > AAA.BBB.CCC.51.www: . ack 1 win 16384 (DF)
20:47:30.017617 193.252.111.151.1920 > AAA.BBB.CCC.51.www: P 1:19(18) ack 1 win 16384 (DF)
20:47:30.018075 AAA.BBB.CCC.51.www > 193.252.111.151.1920: P 1:166(165) ack 19 win 16384
```

```
20:47:30.018079 AAA.BBB.CCC.51.www > 193.252.111.151.1920: F 166:166(0) ack 19 win 16384
20:47:30.184494 193.252.111.151.1920 > AAA.BBB.CCC.51.www: . ack 167 win 16384 (DF)
20:47:30.189302 193.252.111.151.1920 > AAA.BBB.CCC.51.www: F 19:19(0) ack 167 win 16384 (DF)
20:47:30.189384 AAA.BBB.CCC.51.www > 193.252.111.151.1920: . ack 20 win 16384
20:47:30.387892 193.252.111.151.1891 > AAA.BBB.CCC.43.www: S 4213441535:4213441535(0) win 16384 <mss 1452> (DF)
20:47:30.912643 193.252.111.151.1840 > AAA.BBB.CCC.19.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:32.177015 193.252.111.151.1944 > AAA.BBB.CCC.66.www: S 4221365413:4221365413(0) win 16384 <mss 1452> (DF)
20:47:32.246308 193.252.111.151.1946 > AAA.BBB.CCC.67.www: S 4221656578:4221656578(0) win 16384 <mss 1452> (DF)
20:47:32.409428 193.252.111.151.1948 > AAA.BBB.CCC.68.www: S 4222066438:4222066438(0) win 16384 <mss 1452> (DF)
20:47:32.481483 193.252.111.151.1950 > AAA.BBB.CCC.69.www: S 4222398108:4222398108(0) win 16384 <mss 1452> (DF)
20:47:32.584220 193.252.111.151.1952 > AAA.BBB.CCC.70.www: S 4222567089:4222567089(0) win 16384 <mss 1452> (DF)
20:47:32.585187 AAA.BBB.CCC.70.www > 193.252.111.151.1952: S 5855573:5855573(0) ack 4222567090 win 8712 <mss
1460> (DF)
20:47:32.740694 193.252.111.151.1952 > AAA.BBB.CCC.70.www: . ack 1 win 17520 (DF)
20:47:32.751375 193.252.111.151.1952 > AAA.BBB.CCC.70.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:32.943579 AAA.BBB.CCC.70.www > 193.252.111.151.1952: . ack 19 win 8694 (DF)
20:47:33.056713 AAA.BBB.CCC.70.www > 193.252.111.151.1952: P 1:363(362) ack 19 win 8694 (DF)
20:47:33.066435 AAA.BBB.CCC.70.www > 193.252.111.151.1952: F 363:363(0) ack 19 win 8694 (DF)
20:47:33.228668 193.252.111.151.1952 > AAA.BBB.CCC.70.www: . ack 364 win 17520 (DF)
20:47:33.234258 193.252.111.151.1952 > AAA.BBB.CCC.70.www: F 19:19(0) ack 364 win 17520 (DF)
20:47:33.234856 AAA.BBB.CCC.70.www > 193.252.111.151.1952: . ack 20 win 8694 (DF)
20:47:33.288340 193.252.111.151.1965 > AAA.BBB.CCC.70.www: S 4224536176:4224536176(0) win 16384 <mss 1452> (DF)
20:47:33.289082 AAA.BBB.CCC.70.www > 193.252.111.151.1965: S 5983599:5983599(0) ack 4224536177 win 8712 <mss
1460> (DF)
20:47:33.448943 193.252.111.151.1965 > AAA.BBB.CCC.70.www: . ack 1 win 17520 (DF)
20:47:33.455730 193.252.111.151.1965 > AAA.BBB.CCC.70.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:33.458263 AAA.BBB.CCC.70.www > 193.252.111.151.1965: P 1:363(362) ack 19 win 8694 (DF)
20:47:33.458267 AAA.BBB.CCC.70.www > 193.252.111.151.1965: F 363:363(0) ack 19 win 8694 (DF)
20:47:33.630888 193.252.111.151.1965 > AAA.BBB.CCC.70.www: . ack 364 win 17520 (DF)
20:47:33.636055 193.252.111.151.1965 > AAA.BBB.CCC.70.www: F 19:19(0) ack 364 win 17520 (DF)
20:47:33.636621 AAA.BBB.CCC.70.www > 193.252.111.151.1965: . ack 20 win 8694 (DF)
20:47:33.950102 193.252.111.151.1978 > AAA.BBB.CCC.70.www: S 4226320081:4226320081(0) win 16384 <mss 1452> (DF)
20:47:33.950859 AAA.BBB.CCC.70.www > 193.252.111.151.1978: S 5983608:5983608(0) ack 4226320082 win 8712 <mss
1460> (DF)
20:47:34.110622 193.252.111.151.1978 > AAA.BBB.CCC.70.www: . ack 1 win 17520 (DF)
20:47:34.124546 193.252.111.151.1978 > AAA.BBB.CCC.70.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:34.128966 AAA.BBB.CCC.70.www > 193.252.111.151.1978: P 1:605(604) ack 66 win 8647 (DF)
20:47:34.128970 AAA.BBB.CCC.70.www > 193.252.111.151.1978: F 605:605(0) ack 66 win 8647 (DF)
20:47:34.302485 193.252.111.151.1978 > AAA.BBB.CCC.70.www: . ack 606 win 17520 (DF)
20:47:34.309867 193.252.111.151.1978 > AAA.BBB.CCC.70.www: F 66:66(0) ack 606 win 17520 (DF)
20:47:34.310425 AAA.BBB.CCC.70.www > 193.252.111.151.1978: . ack 67 win 8647 (DF)
20:47:35.165163 193.252.111.151.1944 > AAA.BBB.CCC.66.www: S 4221365413:4221365413(0) win 16384 <mss 1452> (DF)
20:47:35.239476 193.252.111.151.1946 > AAA.BBB.CCC.67.www: S 4221656578:4221656578(0) win 16384 <mss 1452> (DF)
20:47:35.409165 193.252.111.151.1948 > AAA.BBB.CCC.68.www: S 4222066438:4222066438(0) win 16384 <mss 1452> (DF)
20:47:35.481940 193.252.111.151.1950 > AAA.BBB.CCC.69.www: S 4222398108:4222398108(0) win 16384 <mss 1452> (DF)
20:47:35.928653 193.252.111.151.1840 > AAA.BBB.CCC.19.www: F 19:19(0) ack 1 win 17424 (DF)
20:47:37.919872 193.252.111.151.2011 > AAA.BBB.CCC.99.www: S 4231861422:4231861422(0) win 16384 <mss 1452> (DF)
20:47:37.920631 AAA.BBB.CCC.99.www > 193.252.111.151.2011: R 0:0(0) ack 4231861423 win 0
20:47:38.912541 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:47:39.680239 193.252.111.151.2044 > AAA.BBB.CCC.116.www: S 4236857438:4236857438(0) win 16384 <mss 1452> (DF)
20:47:39.778380 193.252.111.151.2046 > AAA.BBB.CCC.117.www: S 4237029328:4237029328(0) win 16384 <mss 1452> (DF)
20:47:39.779202 AAA.BBB.CCC.117.www > 193.252.111.151.2046: S 189316:189316(0) ack 4237029329 win 8712 <mss
1460> (DF) [tos 0xdd]
20:47:39.868748 193.252.111.151.2048 > AAA.BBB.CCC.118.www: S 4237456431:4237456431(0) win 16384 <mss 1452> (DF)
20:47:39.868960 AAA.BBB.CCC.118.www > 193.252.111.151.2048: R 0:0(0) ack 4237456432 win 0
20:47:39.939650 193.252.111.151.2046 > AAA.BBB.CCC.117.www: . ack 1 win 17520 (DF)
20:47:39.952249 193.252.111.151.2046 > AAA.BBB.CCC.117.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:39.959974 AAA.BBB.CCC.117.www > 193.252.111.151.2046: P 1:278(277) ack 19 win 8694 (DF) [tos 0xdd]
20:47:39.960336 AAA.BBB.CCC.117.www > 193.252.111.151.2046: F 278:278(0) ack 19 win 8694 (DF) [tos 0xdd]
20:47:39.974168 193.252.111.151.2050 > AAA.BBB.CCC.119.www: S 4237683058:4237683058(0) win 16384 <mss 1452> (DF)
20:47:39.974276 AAA.BBB.CCC.119.www > 193.252.111.151.2050: S 29784109:29784109(0) ack 4237683059 win 8712 <mss
1460> (DF)
20:47:40.128064 193.252.111.151.2046 > AAA.BBB.CCC.117.www: . ack 279 win 17520 (DF)
20:47:40.135216 193.252.111.151.2046 > AAA.BBB.CCC.117.www: F 19:19(0) ack 279 win 17520 (DF)
20:47:40.135489 AAA.BBB.CCC.117.www > 193.252.111.151.2046: . ack 20 win 8694 (DF) [tos 0xdd]
20:47:40.141110 193.252.111.151.2050 > AAA.BBB.CCC.119.www: . ack 1 win 17520 (DF)
20:47:40.149093 193.252.111.151.2050 > AAA.BBB.CCC.119.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:40.158825 AAA.BBB.CCC.119.www > 193.252.111.151.2050: P 1:200(199) ack 19 win 8694 (DF)
20:47:40.162614 AAA.BBB.CCC.119.www > 193.252.111.151.2050: F 200:200(0) ack 19 win 8694 (DF)
20:47:40.187063 193.252.111.151.2054 > AAA.BBB.CCC.117.www: S 4238138825:4238138825(0) win 16384 <mss 1452> (DF)
20:47:40.187366 AAA.BBB.CCC.117.www > 193.252.111.151.2054: S 189319:189319(0) ack 4238138826 win 8712 <mss
1460> (DF)
20:47:40.326141 193.252.111.151.2050 > AAA.BBB.CCC.119.www: . ack 201 win 17520 (DF)
20:47:40.334845 193.252.111.151.2050 > AAA.BBB.CCC.119.www: F 19:19(0) ack 201 win 17520 (DF)
20:47:40.334945 AAA.BBB.CCC.119.www > 193.252.111.151.2050: . ack 20 win 8694 (DF)
20:47:40.350416 193.252.111.151.2054 > AAA.BBB.CCC.117.www: . ack 1 win 17520 (DF)
20:47:40.358049 193.252.111.151.2054 > AAA.BBB.CCC.117.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:40.359191 AAA.BBB.CCC.117.www > 193.252.111.151.2054: P 1:278(277) ack 19 win 8694 (DF)
20:47:40.359489 AAA.BBB.CCC.117.www > 193.252.111.151.2054: F 278:278(0) ack 19 win 8694 (DF)
20:47:40.402620 193.252.111.151.2060 > AAA.BBB.CCC.119.www: S 4239042285:4239042285(0) win 16384 <mss 1452> (DF)
20:47:40.402721 AAA.BBB.CCC.119.www > 193.252.111.151.2060: S 29784179:29784179(0) ack 4239042286 win 8712 <mss
1460> (DF)
20:47:40.529807 193.252.111.151.2054 > AAA.BBB.CCC.117.www: . ack 279 win 17520 (DF)
20:47:40.534677 193.252.111.151.2054 > AAA.BBB.CCC.117.www: F 19:19(0) ack 279 win 17520 (DF)
```

```
20:47:40.534933 AAA.BBB.CCC.117.www > 193.252.111.151.2054: . ack 20 win 8694 (DF)
20:47:40.564304 193.252.111.151.2060 > AAA.BBB.CCC.119.www: . ack 1 win 17520 (DF)
20:47:40.572647 193.252.111.151.2060 > AAA.BBB.CCC.119.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:40.575053 AAA.BBB.CCC.119.www > 193.252.111.151.2060: P 1:200(199) ack 19 win 8694 (DF)
20:47:40.575391 AAA.BBB.CCC.119.www > 193.252.111.151.2060: F 200:200(0) ack 19 win 8694 (DF)
20:47:40.741298 193.252.111.151.2060 > AAA.BBB.CCC.119.www: . ack 201 win 17520 (DF)
20:47:40.746581 193.252.111.151.2060 > AAA.BBB.CCC.119.www: F 19:19(0) ack 201 win 17520 (DF)
20:47:40.746675 AAA.BBB.CCC.119.www > 193.252.111.151.2060: . ack 20 win 8694 (DF)
20:47:40.761048 193.252.111.151.2064 > AAA.BBB.CCC.117.www: S 4239571002:4239571002(0) win 16384 <mss 1452> (DF)
20:47:40.761329 AAA.BBB.CCC.117.www > 193.252.111.151.2064: S 189319:189319(0) ack 4239571003 win 8712 <mss
1460> (DF)
20:47:40.923982 193.252.111.151.2064 > AAA.BBB.CCC.117.www: . ack 1 win 17520 (DF)
20:47:40.934737 193.252.111.151.2064 > AAA.BBB.CCC.117.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:40.956732 193.252.111.151.2066 > AAA.BBB.CCC.119.www: S 4240000972:4240000972(0) win 16384 <mss 1452> (DF)
20:47:40.956837 AAA.BBB.CCC.119.www > 193.252.111.151.2066: S 29784188:29784188(0) ack 4240000973 win 8712 <mss
1460> (DF)
20:47:41.058154 AAA.BBB.CCC.117.www > 193.252.111.151.2064: P 1:183(182) ack 66 win 8647 (DF)
20:47:41.100221 AAA.BBB.CCC.117.www > 193.252.111.151.2064: FP 183:606(423) ack 66 win 8647 (DF)
20:47:41.143006 193.252.111.151.2066 > AAA.BBB.CCC.119.www: . ack 1 win 17520 (DF)
20:47:41.153883 193.252.111.151.2066 > AAA.BBB.CCC.119.www: P 1:66(65) ack 1 win 17520 (DF)
20:47:41.156375 AAA.BBB.CCC.119.www > 193.252.111.151.2066: P 1:605(604) ack 66 win 8647 (DF)
20:47:41.156485 AAA.BBB.CCC.119.www > 193.252.111.151.2066: F 605:605(0) ack 66 win 8647 (DF)
20:47:41.284255 193.252.111.151.2064 > AAA.BBB.CCC.117.www: . ack 607 win 17097 (DF)
20:47:41.289307 193.252.111.151.2064 > AAA.BBB.CCC.117.www: F 66:66(0) ack 607 win 17520 (DF)
20:47:41.289684 AAA.BBB.CCC.117.www > 193.252.111.151.2064: . ack 67 win 8647 (DF)
20:47:41.354235 193.252.111.151.2066 > AAA.BBB.CCC.119.www: . ack 606 win 17520 (DF)
20:47:41.359461 193.252.111.151.2066 > AAA.BBB.CCC.119.www: F 66:66(0) ack 606 win 17520 (DF)
20:47:41.359551 AAA.BBB.CCC.119.www > 193.252.111.151.2066: . ack 67 win 8647 (DF)
20:47:41.678274 193.252.111.151.2069 > AAA.BBB.CCC.125.www: S 4240471574:4240471574(0) win 16384 <mss 1452> (DF)
20:47:41.690651 AAA.BBB.CCC.125.www > 193.252.111.151.2069: S 601293056:601293056(0) ack 4240471575 win 16384
<mss 1452> (DF)
20:47:41.855780 193.252.111.151.2069 > AAA.BBB.CCC.125.www: . ack 1 win 17424 (DF)
20:47:41.864188 193.252.111.151.2069 > AAA.BBB.CCC.125.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:42.404201 193.252.111.151.2082 > AAA.BBB.CCC.131.www: S 4242435736:4242435736(0) win 16384 <mss 1452> (DF)
20:47:42.404378 AAA.BBB.CCC.131.www > 193.252.111.151.2082: S 1474195534:1474195534(0) ack 4242435737 win 8192
<mss 1460>
20:47:42.566255 193.252.111.151.2082 > AAA.BBB.CCC.131.www: . ack 1 win 17520 (DF)
20:47:42.573944 193.252.111.151.2082 > AAA.BBB.CCC.131.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:42.574354 AAA.BBB.CCC.131.www > 193.252.111.151.2082: P 1:119(118) ack 19 win 8192
20:47:42.574453 AAA.BBB.CCC.131.www > 193.252.111.151.2082: F 119:119(0) ack 19 win 8192
20:47:42.679843 193.252.111.151.2044 > AAA.BBB.CCC.116.www: S 4236857438:4236857438(0) win 16384 <mss 1452> (DF)
20:47:42.738340 193.252.111.151.2082 > AAA.BBB.CCC.131.www: . ack 120 win 17520 (DF)
20:47:42.755789 193.252.111.151.2082 > AAA.BBB.CCC.131.www: F 19:19(0) ack 120 win 17520 (DF)
20:47:42.755894 AAA.BBB.CCC.131.www > 193.252.111.151.2082: . ack 20 win 8192
20:47:42.805376 193.252.111.151.2089 > AAA.BBB.CCC.131.www: S 4243568524:4243568524(0) win 16384 <mss 1452> (DF)
20:47:42.805523 AAA.BBB.CCC.131.www > 193.252.111.151.2089: S 1474437161:1474437161(0) ack 4243568525 win 8192
<mss 1460>
20:47:42.849747 193.252.111.151.2069 > AAA.BBB.CCC.125.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:42.964624 193.252.111.151.2089 > AAA.BBB.CCC.131.www: . ack 1 win 17520 (DF)
20:47:42.973086 193.252.111.151.2089 > AAA.BBB.CCC.131.www: P 1:19(18) ack 1 win 17520 (DF)
20:47:42.973436 AAA.BBB.CCC.131.www > 193.252.111.151.2089: P 1:119(118) ack 19 win 8192
20:47:42.973531 AAA.BBB.CCC.131.www > 193.252.111.151.2089: F 119:119(0) ack 19 win 8192
20:47:43.137802 193.252.111.151.2089 > AAA.BBB.CCC.131.www: . ack 120 win 17520 (DF)
20:47:43.145123 193.252.111.151.2089 > AAA.BBB.CCC.131.www: F 19:19(0) ack 120 win 17520 (DF)
20:47:43.145228 AAA.BBB.CCC.131.www > 193.252.111.151.2089: . ack 20 win 8192
20:47:44.851353 193.252.111.151.2069 > AAA.BBB.CCC.125.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:48.849908 193.252.111.151.2069 > AAA.BBB.CCC.125.www: P 1:19(18) ack 1 win 17424 (DF)
20:47:53.868666 193.252.111.151.2069 > AAA.BBB.CCC.125.www: F 19:19(0) ack 1 win 17424 (DF)
20:47:54.779822 193.252.111.151.2226 > AAA.BBB.CCC.201.www: S 4263985530:4263985530(0) win 16384 <mss 1452> (DF)
20:47:54.909881 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:47:56.849331 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:47:57.782056 193.252.111.151.2226 > AAA.BBB.CCC.201.www: S 4263985530:4263985530(0) win 16384 <mss 1452> (DF)
20:48:02.062779 193.252.111.151.2319 > AAA.BBB.CCC.248.www: S 4277418712:4277418712(0) win 16384 <mss 1452> (DF)
20:48:02.249299 193.252.111.151.2323 > AAA.BBB.CCC.250.www: S 4277827328:4277827328(0) win 16384 <mss 1452> (DF)
20:48:05.053381 193.252.111.151.2319 > AAA.BBB.CCC.248.www: S 4277418712:4277418712(0) win 16384 <mss 1452> (DF)
20:48:05.249779 193.252.111.151.2323 > AAA.BBB.CCC.250.www: S 4277827328:4277827328(0) win 16384 <mss 1452> (DF)
20:48:12.851537 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:48:26.912356 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:48:44.844995 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:49:30.904026 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:49:48.843330 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:50:34.906882 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:50:52.877140 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:51:38.894888 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:51:56.835465 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:52:42.889760 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:53:00.830407 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:53:46.888680 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:54:04.824268 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:54:50.882829 193.252.111.151.1840 > AAA.BBB.CCC.19.www: FP 1:19(18) ack 1 win 17424 (DF)
20:55:08.820332 193.252.111.151.2069 > AAA.BBB.CCC.125.www: FP 1:19(18) ack 1 win 17424 (DF)
20:55:54.872060 193.252.111.151.1840 > AAA.BBB.CCC.19.www: R 20:20(0) ack 1 win 17424 (DF)
20:56:12.811660 193.252.111.151.2069 > AAA.BBB.CCC.125.www: R 20:20(0) ack 1 win 17424 (DF)
```

snort output:

```
[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:27.300750 193.252.111.151:1885 -> AAA.BBB.CCC.30:80
TCP TTL:51 TOS:0x0 ID:45341 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFB135C96  Ack: 0x5051471  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:28.227865 193.252.111.151:1901 -> AAA.BBB.CCC.39:80
TCP TTL:51 TOS:0x0 ID:45370 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFB35CBD5  Ack: 0x5FF28D24  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:29.515614 193.252.111.151:1910 -> AAA.BBB.CCC.42:80
TCP TTL:51 TOS:0x0 ID:45415 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFB4E84CD  Ack: 0x32A68  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:34.124546 193.252.111.151:1978 -> AAA.BBB.CCC.70:80
TCP TTL:51 TOS:0x0 ID:45555 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFBE886D2  Ack: 0x5B4D79  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:40.934737 193.252.111.151:2064 -> AAA.BBB.CCC.117:80
TCP TTL:51 TOS:0x0 ID:45747 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFCB2B83B  Ack: 0x2E388  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:41.153883 193.252.111.151:2066 -> AAA.BBB.CCC.119:80
TCP TTL:51 TOS:0x0 ID:45756 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFCB947CD  Ack: 0x1C6787D  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0A 0A  c+dir HTTP/1.0..
0A                                               .

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

reverse lookup:

```
$ dig -x 193.252.111.151

; <<>> DiG 8.3 <<>> -x
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4
;; flags: qr aa ra; QUERY: 1, ANSWER: 1, AUTHORITY: 5, ADDITIONAL: 5
;; QUERY SECTION:
;;      151.111.252.193.in-addr.arpa, type = ANY, class = IN
```

```
;; ANSWER SECTION:
151.111.252.193.in-addr.arpa.  2D IN PTR  APh-Aug-101-1-3-151.abo.wanadoo.fr.

;; AUTHORITY SECTION:
111.252.193.in-addr.arpa.  2D IN NS  ns.wanadoo.fr.
111.252.193.in-addr.arpa.  2D IN NS  ns.wanadoo.com.
111.252.193.in-addr.arpa.  2D IN NS  ns2.wanadoo.fr.
111.252.193.in-addr.arpa.  2D IN NS  ns2.wanadoo.com.
111.252.193.in-addr.arpa.  2D IN NS  ns.ripe.net.

;; ADDITIONAL SECTION:
ns.wanadoo.fr.          1D IN A         193.252.19.10
ns.wanadoo.com.         1D IN A         194.51.131.227
ns2.wanadoo.fr.         1D IN A         193.252.19.11
ns2.wanadoo.com.        1D IN A         194.51.238.2
ns.ripe.net.            5h43m6s IN A    193.0.0.193
```

whois -h whois.ripe.net 193.252.111.151 :

```
% Rights restricted by copyright. See http://www.ripe.net/ripencc/pub-services/db/copyright.html

inetnum:     193.252.111.0 - 193.252.111.255
netname:     IP2000-ADSL-BAS
descr:       France Telecom IP2000 ADSL BAS
country:     FR
admin-c:     ML2808-RIPE
tech-c:      ML2808-RIPE
status:      ASSIGNED PA
remarks:     for hacking, spamming or security problems send ALSO mail to
remarks:     postmaster@wanadoo.fr AND abuse@wanadoo.fr
remarks:     for ANY problem send mail to gestionip.ft@francetelecom.fr
notify:      gestionip.ft@francetelecom.fr
mnt-by:      FT-BRX
changed:     gestionip.ft@francetelecom.fr 20000911
source:      RIPE

route:       193.252.104.0/21
descr:       France Telecom
descr:       FTI
origin:      AS3215
mnt-by:      FT-BRX
changed:     gestionip.ft@francetelecom.fr 20001018
source:      RIPE

person:      Marc Lapeyre
address:     France Telecom
address:     BRX/DRSA/UDS/PSR/TNI
address:     Branche Reseaux
address:     8 rue du Dauphine
address:     69424 Lyon Cedex 03
phone:       +33 4 78 63 89 61
fax-no:      +33 4 78 63 83 74
e-mail:      marc.lapeyre@francetelecom.com
nic-hdl:     ML2808-RIPE
mnt-by:      FT-BRX
changed:     noc@rain.fr 19990205
changed:     gestionip.ft@francetelecom.fr 20000428
changed:     gestionip.ft@francetelecom.com 20010116
source:      RIPE
```


1.2.1. Source of Trace

Our network. Snort box that is connected to a SPAN monitoring port on one of our switches. The switch pipes traffic it receives from some of our border routers into that span port so that snort can see it. See also chapter 2.2. for a more detailed explanation of this setup.

1.2.2. Detect was generated by

Snort 1.7 with full ruleset that I downloaded on 2/26/01. I am also running a tcpdump with –w option to capture tcpdump traffic. Shadow or similar systems are not setup (yet). whois output was generated by whois-ripe.

1.2.3. Snort log format

```
[**] spp_http_decode: IIS Unicode attack detected [**]
03/17-20:47:40.934737 193.252.111.151:2064 -> AAA.BBB.CCC.117:80
TCP TTL:51 TOS:0x0 ID:45747 IpLen:20 DgmLen:105 DF
***AP*** Seq: 0xFCB2B83B  Ack: 0x2E388  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 61 66 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%af../winnt/sy
```

"The first line gives a description of the detect.

The next line gives the timestamp for the detect, the source IP address and port for the packet, and the destination IP address and port […]

For TCP packets, the Time-to-Live is next, along with the hex Type-of-Service value and the IP ID. The DF refers to the request that this packet not be fragmented, a common practice to avoid performance-robbing fragmentation. The TCP flag fields are listed next; a letter indicates that the particular flag is turned on, an asterisk indicates that flag is turned off. The TCP sequence numbers are listed next, along with the tcp window […]

The following lines are the content of the packet. The left hand side shows the hex representation, the right hand side gives the ascii representation for printable characters only. Decimal points are used when the character is unprintable." [STEARNS]

1.2.4. Probability source address was spoofed

Nil. Attacker needs to establish full TCP connection. In order to go through the initial Three-way-handshake, attacking IP has to be the "real" IP. Would not make any sense to spoof (-> redirect responses of the victim) the IP address, since the attacker would not be able to see the responses.

1.2.5. Description of Attack

The so-called "IIS-Unicode-Attack" essentially exploits a vulnerability in IIS 4.0 and IIS 5.0 web servers that "allows remote attackers to read documents outside of the web root, and possibly execute arbitrary commands, via malformed URLs that contain UNICODE encoded characters, aka the *Web Server Folder Traversal* vulnerability." [XFORCE-UNICODE]
CVE number: CVE-2000-0884.

1.2.6. Attack mechanism

Again X-force:
"This vulnerability is a variation on the common 'dot dot' directory traversal attack. Older Web servers were vulnerable to this attack because affected Web servers read '..' directories in URLs and allowed attackers to back out of the web root directory. This technique allowed attackers to navigate the file system at will. IIS and most modern Web servers have incorporated security measures to prevent the 'dot dot' attack. These security measures deny all queries to URLs that contain too many leading slashes or '..' characters. The vulnerability described in this advisory bypasses these restrictions by simply substituting a standard '/' with a UNICODE translation of a '/' character or '\'.

By appending the '..' and a UNICODE slash or backslash after a virtual directory with execute permissions, it is possible for an attacker to execute arbitrary commands. Attackers may execute any command via a specially crafted HTTP query. Attackers may then have the ability to manipulate the appearance of the Web site, download sensitive customer data, or upload and install backdoor software.

All queries are processed under the IUSR_machine context. This context is similar to a normal unprivileged account. However, this account is part of the 'Everyone' and 'Users' group, which allows it access to the web directory and most non-administrative functions. Attackers may not directly modify or delete files owned by the Administrator, nor run commands with privilege."
[XFORCE-UNICODE]

The attacker scanned our public segment for open web server / http ports (tcp port 80) . This can be seen in the tcpdump output on the changing target- IP-addresses. Machines that are found to have port 80 open are then attacked using the IIS-Unicode-vulnerability. The attacker tries to execute "c:\winnt\system32\cmd.exe /c dir" to obtain a listing of the current directory to obtain more information about the system for reconnaissance purposes. Unfortunately, I did not get access to the web server logfiles, so I had to believe my IT people who told me that the attacks failed because the machines were patched with the latest Microsoft updates.

Since the attack is trying to initiate conversations with possible http servers, this is a stimulus (trying to get a response).

1.2.7. Correlation

See CVE-2000-0884.

1.2.8. Evidence of active targeting

The attacker did not know which of the machines in the segment were having the http port open, that is because he scanned the whole segment first, then attacked the machines with port 80 open.

1.2.9. Severity

The severity of the attack is a -3.

(Criticality + Lethality) - (System Countermeasures + Network Countermeasures) = Severity

(3 + 3) - (5 + 4) = -3

- Criticality = 3. The web servers are important, but not essential for the network.
- Lethality = 3. The attack is, in case of success, not bringing the machine down or putting it out of order, but the attacker gained valuable information that could lead to focused exploits.
- System Countermeasures = 5. The web servers are patched, so the attack will not work.
- Network Countermeasures = 4. The IDS detected the attack, appropriate actions (informing the provider etc.) can be taken.

1.2.10. Defensive Recommendations

Make sure that your Internet Information Server packages are up-to-date and has the latest patches installed.

1.2.11. Multiple choice test question

What programs did the attacker try to execute on the web servers?

a) format c:
b) dir c:
c) dir
d) they just wanted to have a shell

Correct answer: b). The log shows us that he tried to execute "cmd.exe /c dir", which just executes the "dir" command, without any parameters.

1.3. Incident 2

```
11/26-21:11:05.159278  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.3:31337
11/26-21:11:05.192439  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.10:31337
11/26-21:11:12.479802  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.19:31337
11/26-21:11:12.487765  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.22:31337
11/26-21:11:12.557400  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.36:31337
11/26-21:11:12.557506  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.44:31337
11/26-21:11:12.603742  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.61:31337
11/26-21:11:12.751112  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.100:31337
11/26-21:11:12.767326  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.103:31337
11/26-21:23:31.604214  [**] spp_portscan: PORTSCAN DETECTED from 209.94.199.143 (THRESHOLD
7 connections in 2 seconds) [**]
11/26-21:11:12.844641  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.126:31337
11/26-21:11:12.846634  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.130:31337
11/26-21:11:12.889881  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.139:31337
11/26-21:11:13.129166  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.152:31337
11/26-21:11:13.129360  [**] Back Orifice [**] 209.94.199.143:31338 -> MY.NET.60.149:31337
```

1.3.1. Source of Trace

The SANS snort archive [SNORTARCHIVE]. SnortA9.txt.

1.3.2. Detect was generated by

Snort. Not sure which version of snort and ruleset, was not found in the files.

1.3.3. Probability address was spoofed

Nil. As in the previous example, the attacker needs to see the response packets in order to take appropriate action, therefore the address is not spoofed.

1.3.4. Description of Attack

This network scan searches for ports that are usually open when a system has Back Orifice installed. Back Orifice is a client/server software package which allows remote administration of a Windows workstation/ -server. Port 31337 is the default tcp port available for BO clients to connect to the server. More information about BO can be found at [XFORCE-BO].

1.3.5. Attack mechanism

This is a reconnaissance attempt. If the scan would find any open ports, the attacker would most likely use Back Orifice to invade and compromise the network. The packets sent are crafted, since the source port never changes (which is unusual). The intervals between the connection attempts are very

small, the attacker tries to scan as many IP addresses as possible in a very short timeperiod.

1.3.6. Correlation

BackOrifice scans are very common in the Snort alert logs. Again, see [XFORCE-BO].

1.3.7. Evidence of active targeting

The attacker did not know which hosts have the BO server installed, hence they had to scan for open ports. It was directed at the network, but not at specific hosts in the first place, like in the previous attack I described.

1.3.8. Severity

(Criticality + Lethality) – (System Measures + Network Measures)

(4 + 3) – (5 + 4) = -2

Criticality = 4. There is potential danger to the network, as if there would be any hosts running Back Orifice, they could be compromised, with unpredictable consequences for the rest of the network.

Lethality = 3. This is just a scan, not an active attack. There is potential danger to the machine if it is running BO, however (see remark above).

System Countermeasures = 5. The attacked IP is not running Back Orifice, hence there is nothing going to happen.

Network Countermeasures = 4. The scan was detected, however, it got the chance to actually get into the network and scan machines.

1.3.9. Defensive Recommendations

Never install Back Orifice. Even if the server port (31337) can be changed, I recommend blocking it since many default installations can be caught this way.

1.3.10. Multiple Choice question

Does above scan indicate active targeting?

a) Yes
b) No

The correct answer is No. The attacker has to scan the network in order to find vulnerable machines running Back Orifice.

1.4. Incident 3

```
SnortA20.txt:12/20-15:05:30.479083  [**] External RPC call [**] 148.228.125.215:1754 -> MY.NET.133.16:111
SnortA20.txt:12/20-15:05:30.492407  [**] External RPC call [**] 148.228.125.215:1826 -> MY.NET.133.87:111
SnortA20.txt:12/20-15:05:33.432083  [**] External RPC call [**] 148.228.125.215:1995 -> MY.NET.133.252:111
SnortA20.txt:12/20-15:05:33.433899  [**] External RPC call [**] 148.228.125.215:1997 -> MY.NET.133.254:111
SnortA20.txt:12/20-15:05:33.434043  [**] External RPC call [**] 148.228.125.215:1740 -> MY.NET.133.2:111
SnortA20.txt:12/20-15:05:33.434096  [**] External RPC call [**] 148.228.125.215:1742 -> MY.NET.133.4:111
SnortA20.txt:12/20-15:05:33.478589  [**] External RPC call [**] 148.228.125.215:1942 -> MY.NET.133.199:111
SnortA20.txt:12/20-15:05:33.478686  [**] External RPC call [**] 148.228.125.215:1813 -> MY.NET.133.74:111
SnortA20.txt:12/20-15:05:33.478738  [**] External RPC call [**] 148.228.125.215:1814 -> MY.NET.133.75:111
SnortA20.txt:12/20-15:05:33.485015  [**] External RPC call [**] 148.228.125.215:1842 -> MY.NET.133.103:111
SnortA20.txt:12/20-15:05:33.485064  [**] External RPC call [**] 148.228.125.215:1843 -> MY.NET.133.104:111
SnortA20.txt:12/20-15:05:33.485983  [**] External RPC call [**] 148.228.125.215:1850 -> MY.NET.133.111:111
SnortA20.txt:12/20-15:05:33.495392  [**] External RPC call [**] 148.228.125.215:1884 -> MY.NET.133.141:111
```

1.4.1. Source of Trace

The SANS snort archive [SNORTARCHIVE]. SnortA20.txt

1.4.2. Detect was generated by

Snort. Not sure which version of snort and ruleset, was not found in the files.

1.4.3. Probability address was spoofed

Nil. As in all the previous examples, the attacker needs to see the response packets in order to take appropriate action, therefore the address is not spoofed.

1.4.4. Description of Attack

The attacker seeks response from the portmapper daemon that is running on Unix machines. In this case, they seek response from a portmapper that is responding to requests from the outside network. The portmapper gives detailed information about which RPC services are running on that machine, as long as they are registered with the portmapper. RPC services generally allow to execute arbitrary commands on a Unix machine, transfer files, provide shell / root level access and so forth. Some RPC services are also vulnerable to buffer overflows and Denial Of Service attacks, in which case they would provide root level access to the machine to the attacker if an attack was successful.

1.4.5. Attack mechanism

This is a reconnaissance attempt in the first place. Real attacks can be expected. The scanner walks through the MY.NET-subnet and connects to the TCP port 111 to check whether it responds with the portmap service or not.

1.4.6. Correlation

External RPC calls are very common and can be found in the [SNORTARCHIVE].

1.4.7. Evidence of active targeting

The attacker did not know which hosts have the portmap service running on port 111, hence they had to scan the whole network for responses. Active targeting yes as far as our network is concerned, no as far as targeting specific hosts are concerned.

1.4.8. Severity

(Criticality + Lethality) – (System Measures + Network Measures)

(5 + 3) – (3 + 4) = 1

Criticality = 5. If an attack would be successful, important servers (DNS etc) could be affected.

Lethality = 3. This is just a scan, not an active attack. There is potential danger to the machine if it is running portmap, however since an open port would most likely be followed up on with an active attack on the RPC installation on that system.

System Countermeasures = 3. It is assumed that the latest patches are installed, so attacks will be useless. However, if the portmaper actually responds to the attacker, vital information about running and supported RPC services could be revealed.

Network Countermeasures = 4. The scan was detected, however, it got the chance to actually get into the network and scan machines.

1.4.9. Defensive Recommendations

- filter incoming requests to TCP port 111 on the border router / firewall, so that they will not reach the Unix servers
- let only internal hosts allow access to the RPC services by adding the respective statements into the hosts.deny / hosts.allow files on the Unix servers.
- Limit the RPC services to those that are really needed.
- Make sure that systems and services are on the latest (security-) patch levels

1.4.10. Multiple Choice question

What does the portmap service / daemon reveal?

a) all RPC services running on that machines
b) all RPC services that are explicitly registered with the portmapper
c) IP routing information
d) Information about the portnumbers of telnet, ftp etc.

The correct answer is b). Only RPC services that are registered with the portmapper are being listed.

1.5. Incident 4

```
12/24-02:28:04.270237  [**] Tiny Fragments - Possible Hostile Activity [**] 61.140.75.4 -> MY.NET.1.8
12/24-02:28:04.270980  [**] Tiny Fragments - Possible Hostile Activity [**] 61.140.75.4 -> MY.NET.1.8
12/24-02:28:49.993349  [**] Tiny Fragments - Possible Hostile Activity [**] 61.140.75.4 -> MY.NET.1.8
12/24-02:28:49.993552  [**] Tiny Fragments - Possible Hostile Activity [**] 61.140.75.4 -> MY.NET.1.8
12/24-02:33:08.191098  [**] Tiny Fragments - Possible Hostile Activity [**] 61.140.75.4 -> MY.NET.1.8
```

1.5.1. Source of Trace

The SANS snort archive [SNORTARCHIVE]. SnortA44.txt

1.5.2. Detect was generated by

Snort. Not sure which version of snort and ruleset, was not found in the files.

1.5.3. Probability address was spoofed

Tiny fragments indicate an attempt to get past the firewall filters, a Denial-Of-Service attack or just a network scan. If this is a DoS attempt, it is thoroughly possible that the IP is spoofed since in that case, the attacker does not need to see the responses; he just sends a huge number of packets, hoping that they get to the host and bring them down.

1.5.4. Description of Attack

"With many IP implementations it is possible to impose an unusually small fragment size on outgoing packets. If the fragment size is made small enough to force some of a TCP packet's TCP header fields into the second fragment, filter rules that specify patterns for those fields will not match. If the filtering implementation does not enforce a minimum fragment size, a disallowed packet

might be passed because it didn't hit a match in the filter." [RFC1858]

This would be the case in this trace: the packets are crafted, not the whole header is in the first packet, but merely forced into the next packet, so that IDS's and firewalls are not able to match the header signature with their databases.

1.5.5. Attack mechanism

This is an attempted attack, most likely. The source IP is establishing multiple connections to the same target IP within a short amount of time. Since it gets through the routers to the target IP, it is up to the operating system on that host to handle the attack.

1.5.6. Correlation

[RFC1858], Bugtraq mail archive #19990324

1.5.7. Evidence of active targeting

Yes. The attacker does not do a scan of the network, they aim directly at one IP. If there was a previous reconnaissance is unknown, though likely.

1.5.8. Severity

(Criticality + Lethality) – (System Measures + Network Measures)

(5 + 5) – (5 + 4) = 1

Criticality = 4. This attack can affect the correct functionality of important servers.

Lethality = 3. If successful, it most likely crashes the IP stack of the victim, thereby disabling the victim and disconnecting it from the network.

System Countermeasures = 5. The victim has an up-to-date kernel that can successfully handle the attack. The attack is unsuccessful.

Network Countermeasures = 4. The attack was detected, however, it got the chance to actually get into the network and attach the machine.

1.5.9. Defensive Recommendations

Make sure that the operating system kernels and the services have the latest patches installed. RFC 1858 recommends *"In a router, one can prevent this sort of attack by enforcing certain limits on fragments passing through, namely, that*

the first fragment be large enough to contain all the necessary header information." [RFC1858]

1.5.10. Multiple Choice question

What is a "tiny fragment" per definition?

a) an IP packet with 0 bytes length
b) a TCP packet which does not have the full header in the first packet
c) a TCP packet which has 1 Byte payload

Correct answer: b) "tiny fragmented" TCP packets have only a part of the header in the first packet because not all the header information fits in it. They are crafted packets.

1.6. Incident 5

```
SnortA30.txt:01/12-09:31:41.697088  [**] SNMP public access [**] 128.46.156.231:1030 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:02.380437  [**] SNMP public access [**] 128.46.156.231:1092 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:04.048761  [**] SNMP public access [**] 128.46.156.231:1093 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:04.082561  [**] SNMP public access [**] 128.46.156.231:1093 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:04.134998  [**] SNMP public access [**] 128.46.156.231:1094 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:10.408144  [**] SNMP public access [**] 128.46.156.231:1096 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:10.649334  [**] SNMP public access [**] 128.46.156.231:1097 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:16.978157  [**] SNMP public access [**] 128.46.156.231:1100 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:17.639995  [**] SNMP public access [**] 128.46.156.231:1102 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:19.840132  [**] SNMP public access [**] 128.46.156.231:1104 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:21.340805  [**] SNMP public access [**] 128.46.156.231:1105 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:32.317978  [**] SNMP public access [**] 128.46.156.231:1114 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:32.569345  [**] SNMP public access [**] 128.46.156.231:1116 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:32.861225  [**] SNMP public access [**] 128.46.156.231:1118 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:36.751741  [**] SNMP public access [**] 128.46.156.231:1122 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:38.252377  [**] SNMP public access [**] 128.46.156.231:1122 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:39.755890  [**] SNMP public access [**] 128.46.156.231:1123 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:40.383916  [**] SNMP public access [**] 128.46.156.231:1126 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:53.662295  [**] SNMP public access [**] 128.46.156.231:1135 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:53.709137  [**] SNMP public access [**] 128.46.156.231:1136 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:53.947386  [**] SNMP public access [**] 128.46.156.231:1138 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:32:54.185026  [**] SNMP public access [**] 128.46.156.231:1139 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:32:56.605378  [**] SNMP public access [**] 128.46.156.231:1143 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:32:58.103883  [**] SNMP public access [**] 128.46.156.231:1144 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:01.310430  [**] SNMP public access [**] 128.46.156.231:1146 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:01.726395  [**] SNMP public access [**] 128.46.156.231:1148 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:06.232417  [**] SNMP public access [**] 128.46.156.231:1149 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:07.736631  [**] SNMP public access [**] 128.46.156.231:1150 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:08.324585  [**] SNMP public access [**] 128.46.156.231:1153 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:33:17.654381  [**] SNMP public access [**] 128.46.156.231:1158 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:33:17.968169  [**] SNMP public access [**] 128.46.156.231:1160 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:33:19.274622  [**] SNMP public access [**] 128.46.156.231:1165 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:22.433652  [**] SNMP public access [**] 128.46.156.231:1166 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:25.471696  [**] SNMP public access [**] 128.46.156.231:1178 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:25.937043  [**] SNMP public access [**] 128.46.156.231:1180 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:33:33.241425  [**] SNMP public access [**] 128.46.156.231:1193 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:34:42.314712  [**] SNMP public access [**] 128.46.156.231:1284 -> MY.NET.100.206:161
SnortA30.txt:01/12-09:34:43.505876  [**] SNMP public access [**] 128.46.156.231:1289 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:34:49.711863  [**] SNMP public access [**] 128.46.156.231:1292 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:34:53.129991  [**] SNMP public access [**] 128.46.156.231:1295 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:34:54.626666  [**] SNMP public access [**] 128.46.156.231:1295 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:34:56.811030  [**] SNMP public access [**] 128.46.156.231:1307 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:36:12.518020  [**] SNMP public access [**] 128.46.156.231:1341 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:36:19.057303  [**] SNMP public access [**] 128.46.156.231:1345 -> MY.NET.100.99:161
SnortA30.txt:01/12-09:36:19.514687  [**] SNMP public access [**] 128.46.156.231:1348 -> MY.NET.100.143:161
SnortA30.txt:01/12-09:37:27.879346  [**] SNMP public access [**] 128.46.156.231:1371 -> MY.NET.100.99:161
```

1.6.1. Source of Trace

The SANS snort archive [SNORTARCHIVE]. SnortA30.txt

1.6.2. Detect was generated by

Snort. Not sure which version of snort and ruleset, was not found in the files.

1.6.3. Probability address was spoofed

Nil. As in the previous example, the attacker needs to see the response packets in order to take appropriate action, therefore the address is not spoofed.

1.6.4. Description of Attack

The target are devices in the network that support SNMP, have TCP port 161 open and accept SNMP requests by supplying the default community string "Public". If this is successful, the attacker can retrieve valuable status information like IP addresses, environment settings, version of the queried device, routing tables and so forth.

1.6.5. Attack mechanism

Looking at the log output, it seems that the attacker has already learned that the target IP MY.NET.100.206, MY.NET.100.143 and MY.NET.100.99 accept the "Public"- community string, since there are subsequent requests being sent to the victim IPs. This could indicate that the attacker is already requesting information he is interested in. This is not an attack, rather reconnaissance. The "Public"-community only supports read-only access by default, so they can only read from, but not write to the device.

1.6.6. Correlation

See [SANSSNMP] *"SANS ID FAQ: Using SNMP for Reconnaissance"*.

1.6.7. Evidence of active targeting

Definitely. Only the above mentioned IP addresses are targeted. So the attacker knows that a) the device supports SNMP and b) it supports "Public" as community string.

1.6.8. Severity

(Criticality + Lethality) – (System Measures + Network Measures)

(4 + 2) – (2 + 3) = 1

Criticality = 4. The victims may contain vital information about the network topology which may make it easier to launch dangerous attacks later on.

Lethality = 2. The SNMP requests do not harm the machine/device directly, it only has read-only access, so the device is not exposed to immediate danger.

System Countermeasures = 5. The device responds to a) external SNMP requests and b) accepts the default read-only community string.

Network Countermeasures = 3. The attacker was detected, however, it got the chance to actually get into the network and scan machines.

1.6.9. Defensive Recommendations

- block incoming SNMP requests from the outside or at least limit it to IP ranges that would need access
- change the default community string to something "non-default"

1.6.10. Multiple Choice question

The default for the write SNMP community string is

a) Write
b) Private

The correct answer is b). Private is the default community string for write access to an SNMP community. This string should NEVER be used after initial installation of SNMP support on a device.

2. Assignment 2 - Describe the State of Intrusion Detection

2.1. Introduction

The assignment stated the following:

"Write a white paper on any single intrusion detection technology or challenge. You may choose any IDS, IDS technology or approach, or network pattern; or you may choose any attack, reconnaissance technique, denial of service, or exploit that operates across a network or within a host system.
If you choose an IDS, IDS technology or approach, or network pattern:
Your paper should be better than the standard papers in the SANS Intrusion Detection FAQ (http://www.sans.org/newlook/resources/IDFAQ/ID_FAQ.htm). Be certain to state clearly what you are trying to show and then to use a combination of explanations and references to demonstrate your point. The purpose of this exercise is for GCIA candidates to demonstrate a clear understanding of a facet of intrusion detection technology or practice."

After reading some topics in the ID FAQ, I decided to write about this topic:

"A primer on how to setup a solid IDS system solution with Snort"

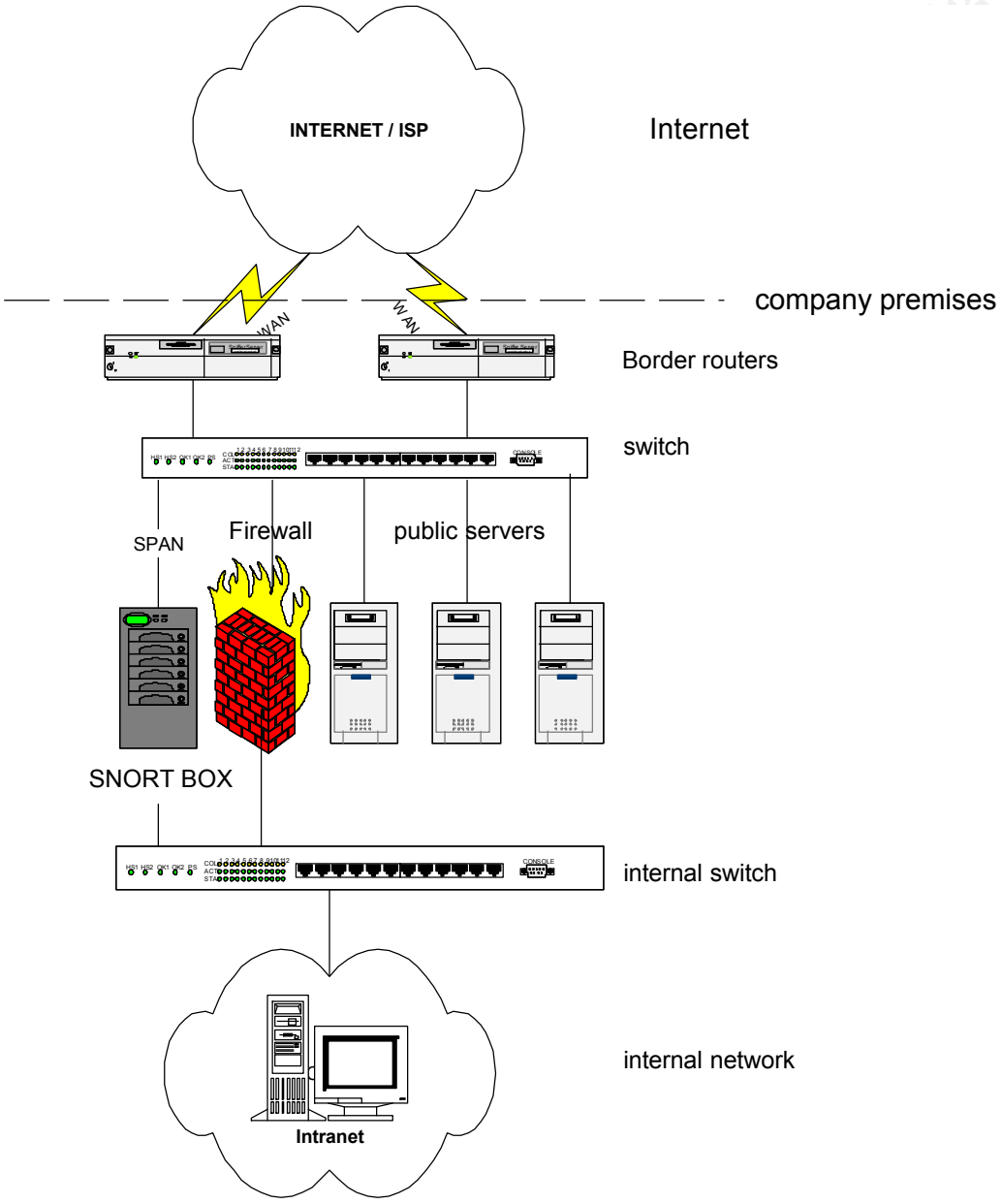
I would like to describe what is in my opinion a good and quick way to get a secure Snort system up and running. The system will be able to analyze traffic, maintain logfiles with SnortSnarf, send syslog messages to a syslog host and defend itself by using portsentry. It will also be able to scan our network for security vulnerabilities.

This chapter will be structured as follows:

- system requirements
- software to install
- system configuration
- network integration

2.2. Integration into the network

2.2.1. Network diagram



2.2.2. Description

Let me describe my setup, it has worked extremely well for me so far

The machine has two NIC/ network interface cards. They will be used as follows:

- NIC1 (eth1) will be connected to the internal network. You will be able to reach the Snort IDS from inside.
- NIC2 (eth0) will be connected to the publicly accessible network, it is plugged in BETWEEN the firewall and the border routers that connect to our Internet Service Provider.

Usually this setup would be very dangerous, since the Snort box would essentially bypass the firewall.

However, the box is connected to a SPAN port on the switch the border routers are connected to. The SPAN port (or Switched Port Analyzer) is e.g. a feature of Cisco catalyst switches. Information can be found at http://www.cisco.com/warp/public/473/41.html#intro

"The SPAN feature was introduced on switches because of a fundamental difference they have with hubs: after a switch learns a MAC address on a port, it forwards traffic for this MAC address directly to the corresponding port. For example, if you want to capture Ethernet traffic sent by host A to host B and both are connected to a hub, just attach a sniffer to this hub as all other ports see the traffic between host A and B. On a switch, after host B's MAC address is learned, unicast traffic from A to B is only forwarded to B's port, and therefore not seen by the sniffer.[…] the sniffer is attached to a port that is configured to receive a copy of every single packet that is sent by host A. This port is called a SPAN port.[…]" [SPAN]

In our scenario, the switch is setup to send copies of traffic to the sniffer/snort box that is exchanged between the border routers and the firewall and the public segment.. meaning it can see ALL the traffic which is coming from the Internet (except for the incoming traffic that is already being blocked/filtered by the routers and not coming through the routers from outside).

However, the span port can NOT be addressed / accessed by traffic coming through the routers! The external NIC of the Snort box cannot be pinged etc from the outside! This is because the "Incoming Packets" option in the switch is set to "disabled". I am not aware of any security flaws (i.e. that that setting can be corrupted / the switch can be corrupted and the box actually be addressed/attacked).

As far as the NIC towards the internal network is concerned: the box is accessible from the internal network, however, only ssh to the box is possible. No other service port is open.

Note: I have Internet connectivity by going through the NIC / eth0 that is plugged into the internal segment.

2.3. System requirements

2.3.1. Hardware

- PC / server which supports/has the following equipment:
  - 2 network interface cards, have to support promiscous mode for packet logging
  - 128 MB RAM
  - standard harddisk (I guess "normal size" is like 20 GB nowadays)
  - CPU Pentium III class, as fast as possible, dual CPU would be awesome for highest performance of the Snort packet sniffer and the SnortSnarf logfile analyzer
  - If accessed locally (with monitor), needs video adapter (type/quality does not matter, up to the person who sets up the system
  - If not accessed locally, console port access should be available in case that the network stack would not work properly

2.3.2. Operating System

- Linux, latest kernel version (2.4.2 as of time of writing)
- Used in this example: Mandrake 7.2 distribution

In order to able to install the latest kernel (or if you upgrade from the 2.2 kernel that came with said Mandrake distro), make sure to read the Documentation/Changes – file in the kernel source tree in order to be able to correctly compile the source.. you will need to upgrade some packages as follows [Source: src/linux/Documentation/Changes]:

"Current Minimal Requirements
==========================
*Upgrade to at *least* these software revisions before thinking you've encountered a bug! If you're unsure what version you're currently running, the suggested command should tell you. Again, keep in mind that this list assumes you are already functionally running a Linux 2.2 kernel. Also, not all tools are necessary on all systems; obviously, if you don't have any PCMCIA (PC*
Card) hardware, for example, you probably needn't concern yourself with pcmcia-cs.

o Gnu C	2.91.66	# gcc --version
o Gnu make	3.77	# make --version
o binutils	2.9.1.0.25	# ld -v
o util-linux	2.10o	# fdformat --version
o modutils	2.4.2	# insmod -V
o e2fsprogs	1.19	# tune2fs
o reiserfsprogs	3.x.0b	# reiserfsck 2>&1|grep reiserfs
o pcmcia-cs	3.1.21	# cardmgr -V
o PPP	2.4.0	# pppd --version
o isdn4k-utils	3.1pre1	# isdnctrl 2>&1|grep version"

The kernel configuration I used is listed at the end of this chapter. The contents are usually stored in the file .default in the src/linux- directory and read upon startup of the kernel configuration tool (make config, make xconfig etc). Please adjust and use if you want to.

Get your kernel sources at: http://rpmfind.net and look for the latest source packages (SRPM) with the "mdk" signature in it, if you are using Mandrake.

Mentioned URL is an excellent source for most of the packages you will need in this documentation.

2.4. The Setup of the System

We want to:	Application name:	Where to get:
log and block intrusion attempts to the Snort box	portsentry	http://www.psionic.com/abacus/portsentry/
run Snort	Snort	http://www.snort.org
analyze and maintain the logfiles	SnortSnarf	http://www.silicondefense.com/software/snortsnarf/
look at the traffic while running Snort	tcpdump	Install with Mandrake installation
scan for security vulnerabilities in our network	NMAP Nessus	**NMAP**: http://www.insecure.org/nmap **Nessus**: http://www.nessus.org
watch our system log	swatch	http://rpmfind.net/linux/rpm2html/search.php?query=swatch
Get quick overview over connections	Sniffit whois	**Sniffit**: http://rpmfind.net/linux/rpm2html/search.php?query=sniffit **whois**: Install with Mandrake installation (whois-ripe)

2.4.1. Configuration of the Snort box

Install all packages as mentioned in the respective manuals.

Configuration specifics are mentioned for each package.

2.4.1.1. Snort

Make sure you download the latest ruleset from http://www.snort.org. I do not have any specific settings to recommend. Just make sure that you only use the rules you want to use. I actually use all the rules, I guess I am being paranoid or trying to keep myself busy ☺.

Run it per

```
snort –sdv –i eth1 –c /etc/snort/snort.rules –l <LOGDIR> -A full
```

this will run Snort with these options:

-d	Dump the Application Layer
-v	Be verbose
-s	Log alert messages to syslog
-i eth1	Listen on interface eth1
–c /etc/snort/snort.rules	use that rules file
-l <LOGIDR>	Log to directory <LOGDIR>
-A	Set alert mode: full mode.. Full mode does normal "classic Snort"-style alerts to the alert file.

2.4.1.2. SnortSnarf

I would recommend to run SnortSnarf on the logfiles per cronjob, like 0600am every morning so that you have the latest happenings right there when you have your first coffee ☺

2.4.1.3. Tcpdump

run it per `tcpdump –i eth1 –w <LOGDIR>/<FILE>` when you start snort. This enables you to have a full tcpdump trace at hand whenever you need to track down an attacker. –w logs the tcpdump packets in binary format to <FILE>.

2.4.1.4. Sniffit

Once in a while, run `sniffit –i –F eth1` and look at the established connections at that time.

2.4.1.5. Swatch

configure it as you like. You might want to highlight log messages generated by

Snort (by default logged per user-facility, level auth (user.auth, user.err)).. see man 5 syslog.conf for details on facilities and log levels.

2.4.1.6. NMAP

configure it as you like.

2.4.1.7. Nessus

configure it as you like.

2.4.1.8. Syslog

given that Snort logs as "user.auth":

```
syslog.conf
user.auth   /var/log/messages
```

If you want to be able to log to a remote loghost:

run syslogd with the "-r" option.

`man syslogd`:

"-r This option will enable the facility to receive a message from the network using an internet domain socket with the syslog service (see services(5)). The default is to not receive any messages from the network."

Important: in order to be able to SEND messages to a remote host, you will have to use the –r option, too, to enable syslogd to connect to the remote syslogger!

change syslog.conf like this:

```
user.auth   @remote.sysloghost
```

This tells syslog to send Snort log messages to the remote.sysloghost.

I have configured Snort to send messages to Micromuse Netcool ®, so it pops up on our main screen in the NOC.. that is very useful.

2.4.1.9. portsentry

This is the portsentry.conf I am using, I deleted comments and unused options. Please see the manual for extensive documentation. In this setup, portsentry essentially monitors common, unused ports (TCP and UDP), gets triggered upon connection attempt to either one of them, drops the route to the attacking host and syslogs that event. Note: this is only running on the internal interface, since the external interface is blocked by the SPAN session setup as described earlier.

```
# PortSentry Configuration
#######################
# Port Configurations #
#######################
#
TCP_PORTS="1,11,15,22,79,111,119,143,540,635,1080,1524,2000,5742,6667,12345,12346,20034,31337,32771,32772,32773,
32774,40421,49724,54320"
UDP_PORTS="1,7,9,69,161,162,513,635,640,641,700,32770,32771,32772,32773,32774,31337,54321"

##########################################
# Advanced Stealth Scan Detection Options #
##########################################
#
ADVANCED_PORTS_TCP="1023"
ADVANCED_PORTS_UDP="1023"

# Default TCP ident and NetBIOS service
ADVANCED_EXCLUDE_TCP="113"
# Default UDP route (RIP), NetBIOS, bootp broadcasts.
ADVANCED_EXCLUDE_UDP="520,138,137,67"

######################
# Configuration Files#
######################
#
# Hosts to ignore
IGNORE_FILE="/usr/local/portsentry/portsentry.ignore"
# Hosts that have been denied (running history)
HISTORY_FILE="/usr/local/portsentry/portsentry.history"
# Hosts that have been denied this session only (temporary until next restart)
BLOCKED_FILE="/usr/local/portsentry/portsentry.blocked"
###################
# Response Options#
###################
#################
# Ignore Options #
#################
BLOCK_UDP="1"
BLOCK_TCP="1"
###################
# Dropping Routes:#
###################
# Newer versions of Linux support the reject flag now. This
# is cleaner than the above option.
KILL_ROUTE="/sbin/route add -host $TARGET$ reject"

###############
# TCP Wrappers#
###############
KILL_HOSTS_DENY="ALL: $TARGET$"


####################
# Scan trigger value#
####################
SCAN_TRIGGER="0"
```

2.4.1.10. whois

To look up networks associated to given IP addresses, start with `whois -h whois.arin.net` and work your way through the country-specific registris (whois.registro.br, whois.ripe.net, whois.apnic.net etc.)

2.4.1.11. System setup

system configuration files:

/etc/resolv.conf - use internal DNS for lookup
/etc/hosts.allow - only allow internal segment access to the machine
/etc/hosts.deny - ALL: ALL,
see `man 5 hosts_access` for details on hosts.* files

services:

only sshd on port 22/tcp is running. As far as access to the webpages generated by SnortSnart is concerned: I would recommend running SnortSnarf, then copying the generated files per secure shell / secure copy to a webserver of your choice.

2.5. Summary

Above I described how my IDS is configured and integrated into the system. Although by far not perfect, I think it is a good starting point.

2.6. Kernel configuration file

Note: you may want to load this file into your kernel configuration tool. Please see the kernel documentation and the configuration tool for help on these options. The kernel (made with `make bzImage` for size purposes) is pretty cleaned up as far as support is concerned, I do not have stuff like sound support compiled into it.

```
# snort.kernel.conf
# only used options are shown
CONFIG_X86=y
CONFIG_ISA=y
CONFIG_UID16=y

#
# Loadable module support
#
CONFIG_MODULES=y
CONFIG_MODVERSIONS=y
CONFIG_KMOD=y

#
# Processor type and features
#
CONFIG_MPENTIUMIII=y
CONFIG_X86_WP_WORKS_OK=y
CONFIG_X86_INVLPG=y
CONFIG_X86_CMPXCHG=y
CONFIG_X86_BSWAP=y
CONFIG_X86_POPAD_OK=y
CONFIG_X86_L1_CACHE_SHIFT=5
CONFIG_X86_TSC=y
CONFIG_X86_GOOD_APIC=y
CONFIG_X86_PGE=y
CONFIG_X86_USE_PPRO_CHECKSUM=y
CONFIG_NOHIGHMEM=y
```

```
CONFIG_SMP=y
CONFIG_HAVE_DEC_LOCK=y

#
# General setup
#
CONFIG_NET=y
CONFIG_X86_IO_APIC=y
CONFIG_X86_LOCAL_APIC=y
CONFIG_PCI=y
CONFIG_PCI_GOANY=y
CONFIG_PCI_BIOS=y
CONFIG_PCI_DIRECT=y
CONFIG_PCI_NAMES=y
CONFIG_HOTPLUG=y

#
# PCMCIA/CardBus support
#
CONFIG_SYSVIPC=y
CONFIG_SYSCTL=y
CONFIG_KCORE_ELF=y
CONFIG_BINFMT_AOUT=m
CONFIG_BINFMT_ELF=y
CONFIG_BINFMT_MISC=y
CONFIG_PM=y

#
# Plug and Play configuration
#
CONFIG_PNP=y
CONFIG_ISAPNP=y

#
# Block devices
#
CONFIG_BLK_DEV_FD=y

#
# Networking options
#
CONFIG_PACKET=y
CONFIG_NETLINK=y
CONFIG_RTNETLINK=y
CONFIG_NETLINK_DEV=y
CONFIG_NETFILTER=y
CONFIG_FILTER is not set
CONFIG_UNIX=y
CONFIG_INET=y
CONFIG_SYN_COOKIES=y

#
# ATA/IDE/MFM/RLL support
#
CONFIG_IDE=y

#
# SCSI support
#
CONFIG_SCSI=y

#
# SCSI support type (disk, tape, CD-ROM)
#
CONFIG_BLK_DEV_SD=y
CONFIG_SD_EXTRA_DEVS=40
CONFIG_BLK_DEV_SR=y
CONFIG_SR_EXTRA_DEVS=2

#
# Some SCSI devices (e.g. CD jukebox) support multiple LUNs
#
CONFIG_SCSI_DEBUG_QUEUES=y
CONFIG_SCSI_MULTI_LUN=y
CONFIG_SCSI_CONSTANTS=y

#
# SCSI low-level drivers
#
CONFIG_SCSI_AIC7XXX=y
CONFIG_AIC7XXX_CMDS_PER_DEVICE=8
CONFIG_AIC7XXX_RESET_DELAY=5
CONFIG_SCSI_SYM53C8XX=y
CONFIG_SCSI_NCR53C8XX_DEFAULT_TAGS=4
```

```
CONFIG_SCSI_NCR53C8XX_MAX_TAGS=32
CONFIG_SCSI_NCR53C8XX_SYNC=20

#
# Network device support
#
CONFIG_NETDEVICES=y

#
# Ethernet (10 or 100Mbit)
#
CONFIG_NET_ETHERNET=y
CONFIG_NET_VENDOR_3COM=y
CONFIG_VORTEX=y

#
# Character devices
#
CONFIG_VT=y
CONFIG_VT_CONSOLE=y
CONFIG_SERIAL=y
CONFIG_SERIAL_CONSOLE=y
CONFIG_UNIX98_PTYS=y
CONFIG_UNIX98_PTY_COUNT=256

#
# Mice
#
CONFIG_MOUSE=y
CONFIG_PSMOUSE=y

#
# Ftape, the floppy tape device driver
#
CONFIG_AGP=y
CONFIG_AGP_INTEL=y
CONFIG_AGP_I810=y
CONFIG_AGP_VIA=y
CONFIG_AGP_AMD=y
CONFIG_AGP_SIS=y
CONFIG_AGP_ALI=y
CONFIG_DRM=y
CONFIG_DRM_TDFX=y
CONFIG_DRM_RADEON=y

#
# File systems
#
CONFIG_AUTOFS4_FS=y
CONFIG_JFFS_FS_VERBOSE=0
CONFIG_ISO9660_FS=y
CONFIG_JOLIET=y
CONFIG_PROC_FS=y
CONFIG_DEVPTS_FS=y
CONFIG_EXT2_FS=y

#
# Network File Systems
#
CONFIG_NFS_FS=y
CONFIG_SUNRPC=y
CONFIG_LOCKD=y

#
# Partition Types
#
CONFIG_MSDOS_PARTITION=y
CONFIG_NLS=y

#
# Native Language Support
#
CONFIG_NLS_DEFAULT="iso8859-1"
CONFIG_NLS_CODEPAGE_437=y
CONFIG_NLS_ISO8859_1=y

#
# Console drivers
#
CONFIG_VGA_CONSOLE=y
CONFIG_VIDEO_SELECT=y

#
# USB support
```

```
#
CONFIG_USB=y


#
# USB Controllers
#
CONFIG_USB_UHCI_ALT=y


#
# USB Device Class drivers
#
CONFIG_USB_STORAGE=y
```

3. Assignment 3 – Analyze This

3.1. Introduction

I have been provided with a month's worth of Snort logfiles, some are missing. In this analysis, I will list the attacks attempted at the GIAC network, the Top 50 attackers and – victims, the source- and destination- IP addresses of the attacks performed and a conclusion with recommendations on how to keep the GIAC network safe and secure.

3.2. List of attacks in ascending order

attack	alerts	sources	dest
STATDX UDP attack	1	1	1
Happy 99 Virus	1	1	1
SITE EXEC - Possible wu-ftpd exploit -GIAC000623	3	3	3
Probable NMAP fingerprint attempt	8	5	6
External RPC call	59	15	25
Back Orifice	77	10	71
TCP SMTP Source Port traffic	100	5	88
Broadcast Ping to subnet 70	154	24	1
SMB Name Wildcard	515	93	171
Russia Dynamo - SANS Flash 28-jul-00	546	2	2
NMAP TCP ping!	558	47	156
SNMP public access	591	20	7
Queso fingerprint	710	52	72
Null scan!	826	527	173
WinGate 1080 Attempt	2239	474	572
Attempted Sun RPC high port access	2257	41	41
Watchlist 000222 NET-NCFC	2401	31	19
connect to 515 from outside	4238	10	2877
Tiny Fragments - Possible Hostile Activity	5340	27	13
DNS udp DoS attack described on unisog	16146	8	6
SYN-FIN scan!	51192	37	27067
Watchlist 000220 IL-ISDNNET-990517	105918	46	100

Total: 194,039 alerts.

3.3. Top 50 of the most attacked machines in the network

# Alerts	victim IP
37604	192.168.201.222
25182	192.168.220.126
9309	192.168.225.234
5411	192.168.1.3
5390	192.168.1.4
5331	192.168.1.5
5181	192.168.202.94
5080	192.168.229.114
4445	192.168.228.214
3148	192.168.1.8
2288	192.168.202.30
1912	192.168.201.130
1517	192.168.130.187
1438	192.168.217.138
1264	192.168.1.10
1221	192.168.98.114
1125	192.168.5.29
1054	192.168.60.11
858	192.168.209.154
803	192.168.213.222
789	192.168.100.230
759	192.168.97.48
727	192.168.217.162
724	192.168.212.38
556	192.168.213.158
540	192.168.6.7
478	192.168.218.14
469	192.168.208.26
450	192.168.219.62
442	194.87.6.38
434	192.168.222.218
430	192.168.207.58
405	192.168.100.209
403	192.168.99.104
396	192.168.214.74
389	192.168.219.250
386	192.168.221.10
340	192.168.217.62
321	192.168.226.174
316	192.168.101.192
278	192.168.253.41
275	192.168.5.29
262	192.168.253.112

259	192.168.130.86
235	192.168.203.190
224	192.168.97.213
221	192.168.223.106
214	192.168.224.138
209	192.168.214.166

3.4. Top 50 of the most active attacking machines

(I looked up the first 10 hosts with whois)

# Alerts	attacker IP	Hostname / whois
48786	212.179.79.2	CREOSCITEX-SIFRA / IL-ISDNNET
17604	211.34.40.1	YOUSUBOOYOUNG-GHS / Korea
9878	195.56.182.206	HU-DATANET-960426 / Hungaria
8565	194.234.48.26	GLOBALONE / Denmark
4096	147.8.182.157	HKUNET / Hong Kong
3052	194.204.224.131	FSC / Casablanca
2353	212.179.77.20	KIBOTZ-SAAR / IL-ISDNNET
1790	200.194.102.99	BRAZIL-BLK2 / RNP Brasil
1273	216.119.15.88	JPS-NETBLK-3 / JPS.NET USA
1054	212.179.58.12	NV-PICTUREVISION / IL-ISDNNET
926	212.179.56.5	
900	159.226.121.37	
713	209.217.166.69	
528	159.226.91.20	
521	202.205.5.10	
469	212.179.58.174	
460	202.101.43.222	
458	61.134.9.133	
442	192.168.205.138	
415	61.140.75.3	
391	202.96.96.3	
389	212.179.67.162	
385	210.12.160.130	
326	202.108.43.152	
286	202.108.43.151	
285	61.140.75.5	
262	192.168.70.38	
243	202.101.43.220	
236	61.140.75.4	
199	61.155.13.3	
178	212.179.17.4	
171	202.101.43.223	
158	159.226.115.1	
137	192.168.70.38	
112	4.4.4.4	

111	209.212.128.47
98	159.226.114.1
97	159.226.39.4
91	207.114.4.46
91	12.77.204.44
85	159.226.228.1
84	63.11.25.117
74	192.168.97.244
71	192.168.162.201
67	204.117.70.5
65	212.72.75.236
62	141.157.104.204
60	192.168.97.155
58	192.168.101.160

3.5. Detailed Description of Attacks Performed

3.5.1. STATDX UDP exploit

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
206.210.80.6	1	9	1	3

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.6.15	1	98	1	15

Just one hit:

```
01/06-06:39:35.583605  [**] STATDX UDP attack [**] 206.210.80.6:10
74 -> MY.NET.6.15:32776
```

Per Snort, this could be an potential attack on the rpc.statd daemon. Discussion can be found in George Bakos excellent practical documentation that can be found at [BAKOS]**:**

"*The rpc.statd is the NFS file lock status reporter. Its function is to track NFS connections with requests to the rpc.lockd. In the event of a server going down, the rpc.statd will attempt to reestablish those locks by communicating the server's status to the NFS client's lock manager. There is a process of the rpc.statd which passes logging information using the syslog() function. The format string passed is user supplied data, with a UID:GID of 0:0, without any proper bounds checking. It is possible, and proven, that this buffer could be overflowed, placing executable code into the process address space and overwriting the process return address, forcing the execution of that code. This is commonly known as "smashing the stack". […]"*

I could not find that 206.x address in any of the scan- or oos- logfiles, so I could not verify if that IP tried to overflow the stack.

3.5.2. Russia Dynamo

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
192.168.205.138	442	442	1	1
38.6.87.194.dynamic.dol.ru	104	104	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
194.87.6.38	442	442	1	1
192.168.205.138	104	108	1	5

```
12/08-15:36:30.735338  [**] Russia Dynamo - SANS Flash 28-jul-00 [**] MY.NET.205.138:6699 -
> 194.87.6.38:2478
12/08-15:36:36.529133  [**] Russia Dynamo - SANS Flash 28-jul-00 [**] MY.NET.205.138:6699 -
> 194.87.6.38:2478
12/08-15:36:54.688783  [**] Russia Dynamo - SANS Flash 28-jul-00 [**] MY.NET.205.138:6699 -
> 194.87.6.38:2478
SnortA26.txt: 12/08-15:37:02.727540  [**] Russia Dynamo - SANS Flash 28-jul-00 [**]
MY.NET.205.138:6699 -> 194.87.6.38:2478
12/08-15:37:04.280071  [**] Russia Dynamo - SANS Flash 28-jul-00 [**] MY.NET.205.138:6699 -
> 194.87.6.38:2478
```

These "alerts" are generated by an inside address establishing connections to an outside address, source port is 6699. 6699 is the default setting if you use the Napster client. I could not find any description on the "Russia Dynamo" in the Internet. I assume that this is a Napster client exchanging data with another Napster user 194.87.6.38 resolves to 38.6.87.194.dynamic.dol.ru, I guess that domain gave the name for this "attack".

Although the destination port, 2478 tcp, is the default port for "SecurSight Authentication Server (SLL)", per snort.org [SNORTPORT]. I found potentially interesting on the GIAC site, reports on 7-29-00 contain the following information: *"[…]Handler on Duty: Stephen Northcutt - Infocon: Yellow - Hottest news is some kind of bad juju with dol.ru, I would consider blocking incoming and OUTGOING traffic to 194.87 or 194.87.6, you will see what I mean as you read through the file. If your system is sending traffic to dol.ru, unplug it! We are particularly interested any outbound activity to dol.ru, and encourage everyone to double-check their logs and let us know.. […]. Hopefully, if people start blocking this address, it will prevent them from getting anymore information and send a strong signal to the Internet Service Provider"* [SANSRUSSIA]

If this is related, I do not know. I would definitely put up a watchlist on that network (see also under recommendations later on.

3.5.3. DNS udp DoS attacked described on unisog

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
futuresite.register.com	16132	16132	3	3
dns3.register.com	4	4	1	1
dns21.register.com	3	3	2	2
dns14.register.com	2	2	2	2
dns2.register.com	2	2	1	1
dns1.register.com	1	1	1	1
dns7.register.com	1	1	1	1
dns16.register.com	1	1	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.1.3	5411	5452	1	13
192.168.1.4	5390	5408	1	10
192.168.1.5	5331	5352	1	8
192.168.1.8	6	3219	4	31
192.168.1.10	6	1279	3	22
192.168.1.9	2	16	2	10

The messages in the alert log files look like this:

```
01/06-18:48:29.945147  [**] DNS udp DoS attack described on unisog [**]
209.67.50.203:14219 -> MY.NET.1.3:53
01/06-18:48:30.659835  [**] DNS udp DoS attack described on unisog [**]
209.67.50.203:17976 -> MY.NET.1.3:53
01/06-18:48:30.699725  [**] DNS udp DoS attack described on unisog [**]
209.67.50.203:13983 -> MY.NET.1.4:53
01/06-18:48:30.711219  [**] DNS udp DoS attack described on unisog [**]
209.67.50.203:27093 -> MY.NET.1.3:53
01/06-18:48:31.734393  [**] DNS udp DoS attack described on unisog [**]
209.67.50.203:12987 -> MY.NET.1.5:53
```

The source IP 209.67.50.203 / futuresite.register.com is the major responsible host for triggering the Snort IDS here. It attempts to establish a LOT of connections (around 530) to port 53/udp within around 30 minutes.
The only reference to this I found in the Neohapsis archives [NEOHAPSIS].This could be related to the attempt to crash Sygate DNS servers: *"[…] As you can see from http://www.sybergen.com/support/fix.htm this issue is most likely still valid and people running Sygate are vulnerable to an annoying exploit that can keep a companies network from "surfing the net" yadda yadda yadda. If your a user of Sygate then send an eMail to support@sygate.com and I am sure if enough people eMail they will write a fix.*
//Sygate Crash by: marc@eeye.com (April-00)
//http://www.eEye.com
//Will crash Sygate (http://www.sygate.com/) when ran from the internal LAN.
//Play with source routing to get it to work across the internet.
//Just hit the Internal IP of the Sygate machine. […]"

3.5.4. SYN-FIN scan

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
211.34.40.1	17604	17604	17604	17604
195.56.182.206	9878	9878	9878	9878
194.234.48.26	8565	8565	8565	8565
147.8.182.157	4096	4096	4096	4096
194.204.224.131	3052	3052	3052	3052
alphac.lnk.telstra.net	1951	1951	1914	1914
200.194.102.99	1790	1790	1790	1790
www.turunhippos.fi	1580	1580	1580	1580
adsl-63-204-152-253.dsl.snfc21.pacbell.net	1242	1242	1242	1242
ABoulogne-102-1-5-9.abo.wanadoo.fr	706	706	706	706
far-rodrigo.cs.technion.ac.il	630	630	630	630
adsl-64-161-240-254.dsl.lsan03.pacbell.net	44	45	41	42
Norchem.DSL.InfoMagic.NET	25	25	25	25
1Cust117.tnt1.yakima.wa.da.uu.net	4	95	4	86
A050-1255.LAUR.splitrock.net	2	2	1	1
A050-0465.LAUR.splitrock.net	2	2	1	1
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.253.112	19	294	17	24
192.168.21.15	8	8	8	8
192.168.11.212	7	8	7	8
192.168.5.125	7	7	7	7
192.168.25.149	6	7	6	7
192.168.1.204	6	6	6	6
192.168.20.199	6	7	6	7
192.168.25.135	6	7	6	7
192.168.1.117	6	6	6	6
192.168.25.72	6	6	6	6
192.168.7.184	6	6	6	6
192.168.223.255	6	6	6	6
192.168.18.77	6	6	6	6
192.168.25.213	6	6	6	6
192.168.232.35	6	6	6	6
192.168.223.133	6	7	6	7
192.168.11.8	6	6	6	6

"This type of scanning uses an impossible flag combination to probe target hosts for listen ing ports. These packets are crafted by a scanning tool, because normal packets will never have both the SYN and FIN bits set simultaneously. This scan can be used to possibly avoid detection of IDS systems that only look for SYN only connections. Also Linux will give a distinct response to a SYN-FIN packet (a SYN-FIN-ACK) that will clue the attacker that the target is a Linux host." [CHL]

3.5.5. Watchlist 000220 IL-ISDNNET-990517

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
212.179.79.2	48786	48786	41	41
clnt-27111.bezeqint.net	39015	39015	2	2
cable-95005.bezeqint.net	4563	4563	4	4
212.179.77.20	2353	2353	2	2
bzq-44-105.bezeqint.net	1517	1517	1	1
fr-c42102.bezeqint.net	1387	1387	2	2
clnt-38135.bezeqint.net	1221	1221	1	1
212.179.58.12	1054	1054	1	1
fr-c27241.bezeqint.net	1002	1002	8	8
212.179.56.5	926	926	3	3
clnt-8164.bezeqint.net	566	566	14	14
clnt-15122.bezeqint.net	478	478	1	1
212.179.58.174	469	469	1	1
clnt-7173.bezeqint.net	453	453	2	2
bzq-125-114.bezeqint.net	429	429	1	1
	# Alerts (sig)	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
Destination-IPs				
192.168.201.222	37604	37609	1	6
192.168.220.126	25182	25183	1	2
192.168.225.234	9309	9314	1	4
192.168.202.94	5181	5253	4	20
192.168.229.114	5080	5082	1	3
192.168.228.214	4445	4448	1	4
192.168.202.30	2288	2291	1	4
192.168.201.130	1912	2047	1	17
192.168.130.187	1517	1520	1	4
192.168.217.138	1438	1447	3	8
192.168.98.114	1221	1230	1	8
192.168.5.29	1125	1429	1	29
192.168.60.11	1054	1457	1	140
192.168.209.154	858	859	4	5
192.168.213.222	803	803	1	1
192.168.97.48	759	763	2	6

Watchlists are put up to monitor networks in the Internet that seem to cause a large number of alerts in IDS's. Two networks infamous for triggering alerts are the Israelian ISDNNET and the The Computer Network Center Chinese Academy of Sciences.

Whois excerpt for the ISDNNET:

```
aut-num:      AS8551
as-name:      ISDN-NET-AS
descr:        Bezeq International
[…]
route:        212.179.0.0/17
descr:        ISDN Net Ltd.
origin:       AS8551
notify:       hostmaster@isdn.net.il
mnt-by:       AS8551-MNT
changed:      hostmaster@isdn.net.il 19990610
source:       RIPE
```

3.5.6. Watchlist 000222 NET-NCFC

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
159.226.121.37	900	900	5	5
159.226.91.20	528	528	1	1
panda.cstnet.net.cn	174	174	1	1
159.226.115.1	158	158	2	2
159.226.114.1	98	98	4	4
159.226.39.4	97	97	2	2
159.226.228.1	85	85	5	5
aphy.iphy.ac.cn	69	69	5	5
159.226.159.50	43	43	3	3
infoc3.icas.ac.cn	32	32	1	1
159.226.63.200	32	32	1	1
159.226.158.188	29	29	1	1
159.226.92.10	25	25	1	1
lsc02.iss.ac.cn	24	24	3	3
159.226.5.222	20	20	1	1
iss01.iss.ac.cn	16	16	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.100.230	789	808	6	15
192.168.6.7	540	551	4	14
192.168.253.41	278	296	8	18
192.168.5.29	275	1429	1	29
192.168.253.42	112	162	6	14
192.168.253.43	103	170	6	14
192.168.253.53	61	64	1	4
192.168.6.35	55	63	3	10
192.168.253.51	43	44	3	4
192.168.6.34	40	48	3	7
192.168.1.2	27	30	3	6
192.168.145.9	26	33	2	7
192.168.253.52	22	24	4	6
192.168.6.47	15	24	1	10
192.168.145.18	11	15	1	5

Whois excerpt for the NCFC:

```
The Computer Network Center Chinese Academy of Sciences (NET-NCFC)
   P.O. Box 2704-10,
   Institute of Computing Technology Chinese Academy of Sciences
   Beijing 100080, China
   CN

   Netname: NCFC
   Netblock: 159.226.0.0 - 159.226.255.255
```

3.5.7. WinGate 1080 Attempt

Source	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
209.212.128.47	111	111	71	71
12.77.204.44	91	91	83	83
207.114.4.46	91	91	74	74
204.117.70.5	67	67	26	26
212.72.75.236	65	65	27	27
199.173.178.2	55	55	31	31
198.63.2.192	52	52	35	35
top100.rambler.ru	51	51	15	15
205.136.57.121	44	44	32	32
philly.pa.us.dal.net	41	41	33	33
tiger.swepipe.com	40	40	26	26
pluto.planetwebspace.com	37	37	1	1
198.92.138.226	37	37	5	5
216.152.64.142	30	30	6	6
finance.webmaster.com	27	27	4	4

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.60.8	134	243	43	115
192.168.208.22	110	112	7	9
192.168.60.11	96	1457	35	140
192.168.60.38	95	154	34	75
192.168.15.178	75	79	37	41
192.168.202.94	54	5253	11	20
192.168.60.16	47	137	23	63
192.168.100.203	41	42	9	10
192.168.201.146	33	35	6	8
192.168.98.169	26	28	8	10
192.168.201.46	25	27	9	11
192.168.225.62	25	26	1	2
192.168.217.146	22	29	5	8
192.168.98.184	19	21	7	9
192.168.98.194	17	19	8	10
192.168.218.218	16	19	5	8

"Port 1080 is a common port for Wingate to operate on. Vulnerabilities with certain versions of WinGate have been found where it allows an attacker to

access the Wingate server hard disk." [CHL]

3.5.8. TCP SMTP Source Port Traffic

Source-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Dsts (sig)**	**# Dsts (total)**
63.11.25.117	84	95	84	86
vismed.nida.nih.gov	11	11	1	1
eu214.st161-net74.ip.superonlinecorporate.com	2	2	1	1
irc.east.gblx.net	2	2	1	1
adsl-64-161-240-254.dsl.lsan03.pacbell.net	1	45	1	42
	# Alerts (sig)	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
Destination-IPs				
192.168.253.42	11	162	1	14
192.168.199.71	2	5	1	4
192.168.5.27	2	3	1	2
192.168.140.67	1	3	1	3
192.168.140.220	1	4	1	3
192.168.140.222	1	3	1	3
192.168.68.29	1	2	1	2
192.168.105.136	1	2	1	2
192.168.151.66	1	3	1	3
192.168.140.155	1	2	1	2
192.168.140.156	1	4	1	4
192.168.71.34	1	3	1	3
192.168.140.191	1	2	1	2
192.168.182.21	1	2	1	2
192.168.2.206	1	5	1	5
192.168.140.34	1	3	1	3

"Both the source port and the destination port is 25 in this alert. The source port is usually one of the higher ephemeral ports. Since the source port is 25, the packet could possibly be crafted." [CHL]

3.5.9. Attempted SUN RPC highport access

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
fes-d004.icq.aol.com	570	570	4	4
fes-d012.icq.aol.com	493	493	4	4
fes-d010.icq.aol.com	398	398	4	4
fes-d008.icq.aol.com	154	154	4	4
fes-d005.icq.aol.com	150	150	2	2
toc-d01.blue.aol.com	91	91	1	1
fes-d006.icq.aol.com	73	73	2	2
fes-d009.icq.aol.com	63	63	4	4
fes-d015.icq.aol.com	59	59	2	2
216.13.244.241	45	45	1	1
cc889103-a.hwrd1.md.home.com	35	35	1	1
fes-d002.icq.aol.com	24	24	1	1
msgr-sb5.msgr.hotmail.com	19	19	1	1
fes-d003.icq.aol.com	14	14	1	1
baltbay1-29.dial.umd.edu	7	7	1	1
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.213.158	556	663	9	19
192.168.222.218	434	439	1	4
192.168.97.213	224	230	1	7
192.168.223.106	221	224	1	4
192.168.224.138	214	215	1	2
192.168.213.158	104	663	7	19
192.168.105.115	90	91	3	4
192.168.97.208	78	81	1	4
192.168.98.238	57	60	2	5
192.168.221.130	45	46	1	2
192.168.97.96	44	46	1	3
192.168.99.51	42	49	2	8
192.168.98.199	19	22	1	4
192.168.226.242	14	17	1	4
192.168.97.245	12	12	1	1

"Remote Procedure Calls (RPC) allow a user (or an attacker) to execute programs on another computer. There have been multiple vulnerabilities reported that have been widely exploited. It is one of SANS Top ten Threats. " [CHL]

Although most of these loggings are most likely false positives! "Attacks" from *.icq.aol.com are most likely standard ICQ connections which usually go from source port 4000 on the ICQ server to ports in the 32xxx range on the client – a range also used for some RPC services, hence the alert. Correlation with the alert file should reveal this.

3.5.10. Broadcast Ping to subnet 70

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
ns.endzone.ro	52	52	1	1
ppp8-2.digiro.net	26	26	1	1
ppp220091.fx.ro	17	17	1	1
ppp220137.fx.ro	12	12	1	1
ppp220238.fx.ro	8	8	1	1
stan.pcnet.ro	6	6	1	1
ppp201.pcnet.ro	4	4	1	1
193.231.169.166	3	3	1	1
www2.powertech.no	3	3	1	1
ppp220125.fx.ro	3	3	1	1
pc7.dwcomp.mediasat.ro	3	3	1	1
ppp-42-208.21-151.libero.it	2	2	1	1
8dial88.xnet.ro	2	2	1	1
cc13991-a.zwoll1.ov.nl.home.com	2	2	1	1
211.33.158.136	2	2	1	1
	# Alerts (sig)	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
Destination-IPs				
192.168.70.255	154	156	24	26

"A broadcast ping is usually performed to receive an ICMP Echo Response from every host on the subnet (particularly subnet 70). If enough of these broadcast pings are sent, a possible ICMP flood could choke the network bandwidth." [CHL]

3.5.11. BackOrifice

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
A030-0650.MLE2.splitrock.net	19	19	3	3
cr518339-a.wlfdle1.on.wave.home.com	16	16	1	1
A010-0680.MLE2.splitrock.net	11	11	2	2
A010-0679.MLE2.splitrock.net	10	10	1	1
A050-0529.LAUR.splitrock.net	9	9	1	1
A030-0665.MLE2.splitrock.net	9	9	1	1
A050-0798.LAUR.splitrock.net	8	8	1	1
cr859517-a.surrey1.bc.wave.home.com	8	13	2	2
A050-1175.LAUR.splitrock.net	7	7	1	1
A030-0655.MLE2.splitrock.net	7	7	1	1
cc99254-a.catv1.md.home.com	6	6	1	1
A020-0553.LAUR.splitrock.net	6	6	1	1
A040-0348.LAUR.splitrock.net	6	6	1	1
A010-0677.MLE2.splitrock.net	6	6	2	2
A050-1124.LAUR.splitrock.net	6	6	2	2
A030-0661.MLE2.splitrock.net	5	5	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.60.11	131	1457	96	140
192.168.60.8	94	243	67	115
192.168.60.16	86	137	37	63
192.168.6.39	62	62	25	25
192.168.6.44	61	66	21	26
192.168.60.38	50	154	36	75
192.168.5.29	25	1429	23	29
192.168.105.120	21	31	2	9
192.168.253.112	13	294	6	24
192.168.180.26	11	16	4	8
192.168.217.242	10	19	10	14
192.168.201.78	9	20	6	11
192.168.253.114	5	19	5	14
192.168.5.10	5	7	5	7
192.168.99.51	5	49	5	8
192.168.253.105	4	8	4	7

"A very common remote administration program used as a trojan horse for Windows hosts. Attacks usually scan for the port BackOrifice listens on, port 31337. If BackOrifice is running on a host, and an attacker finds it, the attacker can connect to it using the Back Orifice client and basically take over the host to

do a countless number of evils." [CHL]

3.5.12. SNMP public access

Source	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
ece156-dhcp-2.ecn.purdue.edu	161	161	3	3
192.168.97.244	74	74	1	1
192.168.162.201	71	71	1	1
192.168.97.155	60	60	1	1
192.168.98.134	40	40	1	1
192.168.97.52	33	33	1	1
192.168.97.168	33	33	1	1
192.168.97.133	26	26	1	1
192.168.111.156	19	36	1	3
192.168.97.186	13	13	1	1
128.183.38.30	12	12	1	1
192.168.97.200	10	10	1	1
192.168.206.222	8	8	1	1
192.168.97.206	7	7	1	1
192.168.97.107	7	7	1	1
	# Alerts (sig)	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
Destination				
192.168.101.192	316	374	14	15
192.168.100.99	104	106	1	3
192.168.50.154	94	94	3	3
192.168.100.143	36	37	1	2
192.168.100.206	21	22	1	2
192.168.154.26	12	14	1	3
192.168.14.1	8	11	1	2

"SNMP consists of two parts: the SNMP manager and the SNMP agent. The SNMP manager retrieves configuration and performance counter information from the agents by issuing "get" commands, and they can also change a network element's configuration by issuing "set" commands. SNMP's authentication is accomplished through the use of a community string. Unfortunately, this community string is often set by default to a predictable string. The most common string for reads is "Public", and the most common string for writes is "Private". Since these read and write community strings are so common, it is easy for an attacker to exploit SNMP. The attacker can do much reconnaissance or change settings in network elements as well. According to SANS Top Ten Threats, default SNMP is number 10." [CHL]

3.5.13. Null Scan

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
A030-0650.MLE2.splitrock.net	19	19	3	3
cr518339-a.wlfdle1.on.wave.home.com	16	16	1	1
A010-0680.MLE2.splitrock.net	11	11	2	2

A010-0679.MLE2.splitrock.net	10	10	1	1
A050-0529.LAUR.splitrock.net	9	9	1	1
A030-0665.MLE2.splitrock.net	9	9	1	1
A050-0798.LAUR.splitrock.net	8	8	1	1
cr859517-a.surrey1.bc.wave.home.com	8	13	2	2
A050-1175.LAUR.splitrock.net	7	7	1	1
A030-0655.MLE2.splitrock.net	7	7	1	1
cc99254-a.catv1.md.home.com	6	6	1	1
A020-0553.LAUR.splitrock.net	6	6	1	1
A040-0348.LAUR.splitrock.net	6	6	1	1
A010-0677.MLE2.splitrock.net	6	6	2	2
A050-1124.LAUR.splitrock.net	6	6	2	2
A030-0661.MLE2.splitrock.net	5	5	1	1
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.60.11	131	1457	96	140
192.168.60.8	94	243	67	115
192.168.60.16	86	137	37	63
192.168.6.39	62	62	25	25
192.168.6.44	61	66	21	26
192.168.60.38	50	154	36	75
192.168.5.29	25	1429	23	29
192.168.105.120	21	31	2	9
192.168.253.112	13	294	6	24
192.168.180.26	11	16	4	8
192.168.217.242	10	19	10	14
192.168.201.78	9	20	6	11
192.168.253.114	5	19	5	14
192.168.5.10	5	7	5	7
192.168.99.51	5	49	5	8
192.168.253.105	4	8	4	7

"This type of scan has none of the tcp flag bits set. Null scans can be used to map out a network.." [CHL]

3.5.14. SMB Name Wildcard

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
141.157.104.204	62	62	1	1
192.168.101.160	58	58	1	1
132.239.165.19	23	23	1	1
192.168.111.156	17	36	2	3
130.54.113.11	16	16	7	7
130.203.237.41	14	14	5	5
rcgpc.gtri.gatech.edu	13	13	5	5
130.153.158.82	11	11	4	4
130.60.57.66	11	11	5	5
130.160.179.100	10	10	5	5
stm.rilab.kyoto-u.ac.jp	10	10	4	4
slip129-37-151-19.on.ca.prserv.net	9	9	1	1
130.91.206.91	9	9	5	5
130.118.152.211	9	9	5	5
130.237.217.108	9	9	5	5
130.39.126.168	8	8	3	3
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.6.15	67	98	2	15
192.168.101.192	58	374	1	15
192.168.98.212	23	25	1	3
192.168.100.130	10	19	2	11
192.168.50.239	10	10	1	1
192.168.100.165	7	38	1	18
192.168.125.41	7	7	1	1
192.168.206.134	6	7	3	4
192.168.4.237	6	6	1	1
192.168.108.35	5	5	1	1
192.168.204.218	5	7	1	3
192.168.228.36	5	7	1	3
192.168.220.37	5	6	1	2
192.168.4.128	5	9	1	5
192.168.106.162	5	7	1	3
192.168.203.230	4	6	1	3

"The SMB Wildcard is a Netbios name query and this is a sign that an attacker is trying to access a share with poor access control. When improperly configured, sharing via Netbios can make sensitive information susceptible to an attacker connected to the network. It is written on the SANS top ten list that 'When file sharing is enabled on Windows machines they become vulnerable to both information theft and certain types of quick-moving viruses'. http://www.sans.org/topten.htm" [CHL]

3.5.15. Queso Fingerprint

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
monitor.dslreports.com	204	204	1	1
192.dsl7839.rcsis.com	144	144	12	12
x05r1b.wh2.tu-dresden.de	49	49	2	2
x07r5c.wh2.tu-dresden.de	41	41	3	3
x14l1a.wh2.tu-dresden.de	30	30	4	4
x08m1a.wh2.tu-dresden.de	26	26	1	1
x09r5b.wh2.tu-dresden.de	23	23	3	3
pool4047.studentenheim.uni-tuebingen.de	21	21	5	5
zz2123.staff.digex.net	19	19	4	4
x09l2a.wh2.tu-dresden.de	15	15	4	4
204.42.254.5	11	11	3	3
x04r4b.wh2.tu-dresden.de	10	10	4	4
64.80.118.241	10	10	6	6
x14m5b.wh2.tu-dresden.de	9	9	1	1
csociety-ftp.ecn.purdue.edu	8	8	1	1
x12r1a.wh2.tu-dresden.de	8	8	2	2
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.219.114	204	205	1	2
192.168.201.130	127	2047	10	17
192.168.201.62	51	55	7	11
192.168.204.38	39	41	1	3
192.168.223.226	38	42	2	6
192.168.224.242	37	39	2	4
192.168.201.66	28	34	7	12
192.168.202.46	20	26	4	8
192.168.53.108	16	17	2	3
192.168.60.8	10	243	1	115
192.168.219.210	8	10	1	3
192.168.253.43	7	170	2	14
192.168.53.184	6	6	4	4
192.168.201.78	6	20	1	11
192.168.210.6	6	7	1	2
192.168.60.38	6	154	2	75

"Queso is a tool that determines the operating system of the target host. It does so by analyzing the responses to impossible packets that Queso sends. Queso uses destination port 80 by default but it can use other port that is open on the target host. Once the operating system is determined, an attacker can then use the appropriate exploits on the target host." [CHL]

3.5.16. NMAP TCP Ping

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
192.168.70.38	262	399	130	141
192.102.197.234	55	55	3	3
bestroute2-t.alcatel.fr	46	46	8	8
63.119.91.2	41	41	10	10
non-invasive-proximity-checking-device.safeweb.com	19	19	8	8
64.64.226.2	19	19	6	6
202.187.24.3	16	16	9	9
non-invasive-proximity-checking-device.safeweb.com	8	8	5	5
212.208.74.129	8	8	3	3
lp2.sealedair.com	7	7	4	4
199.197.130.21	6	6	2	2
199.197.135.21	6	6	4	4
199.36.49.2	5	5	3	3
204.155.48.3	4	4	3	3
194.186.36.190	4	4	1	1
2.2.2.2	4	4	3	3
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.1.8	63	3219	8	31
192.168.1.3	38	5452	9	13
192.168.253.125	27	31	16	20
192.168.100.165	27	38	13	18
192.168.60.14	23	24	16	17
192.168.1.5	18	5352	4	8
192.168.110.39	17	18	2	3
192.168.1.4	14	5408	5	10
192.168.100.230	11	808	3	15
192.168.253.114	10	19	5	14
192.168.1.10	6	1279	2	22
192.168.6.34	6	48	2	7
192.168.181.144	5	8	5	8
192.168.0.61	5	5	1	1
192.168.1.9	5	16	2	10
192.168.6.7	4	551	3	14

"NMAP is a versatile and powerful tool to map out a network and to see what services are running on the hosts of the targeted network. The TCP Ping is one of its options to scan what hosts are up on a network as opposed to an ICMP ping." [CHL]

3.5.17. Connect to 515 from inside

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
192.168.70.38	137	399	89	141
192.168.98.151	9	9	1	1

192.168.60.38	3	3	1	1
192.168.253.12	3	3	1	1
192.168.99.244	2	2	1	1
192.168.60.16	1	1	1	1
192.168.163.17	1	1	1	1
192.168.179.78	1	1	1	1
192.168.219.122	1	8	1	2
192.168.219.194	1	1	1	1
	# Alerts (sig)	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
Destination-IPs				
216.181.129.185	9	9	1	1
192.168.0.114	4	7	1	1
192.168.0.22	3	4	1	1
192.168.0.109	3	5	1	1
64.23.4.67	3	3	1	1
192.168.0.30	3	5	1	1
192.168.0.67	3	5	1	1
128.8.3.106	3	3	1	1
192.168.0.108	3	6	1	1
192.168.0.32	3	4	1	1
212.187.65.135	3	3	1	1
192.168.0.1	3	4	1	1
192.168.0.60	3	6	1	1
192.168.0.77	2	2	1	1
129.155.192.99	2	2	1	1
192.168.0.54	2	5	1	1

"Port 515 is where the UNIX LPR and LPRng services run. SANS has reported that are advisories released concerning LPR service vulnerabilities, for many varities of Linux and for the other UNIX BSD varieties. That report is found at: http://www.sans.org/newlook/alerts/port515.htm. An attacker can exploit the string-format vulnerability present in the LPRng service, and potentially gain root access or execute malicious code." [CHL]

3.5.18. Probable NMAP fingerprint attempt

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
cr859517-a.surrey1.bc.wave.home.com	4	13	2	2
130.239.129.109	1	4	1	1
211.109.37.120	1	1	1	1
153.19.144.207	1	1	1	1
ns.isrd.net	1	6	1	2

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.105.120	3	31	1	9
192.168.201.222	1	37609	1	6
192.168.98.154	1	11	1	8
192.168.98.147	1	13	1	6
192.168.209.78	1	6	1	3
192.168.217.146	1	29	1	8

"NMAP can also be used to determine with fair accuracy the operating system of the target host. Once the operating system is determined, an attacker can then use the appropriate exploits on the target host." [CHL]

3.5.19. External RPC Call

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
148.228.125.215	13	13	13	13
jsmala.polmoslancut.com.pl	8	8	8	8
206.210.80.6	8	9	3	3
1Cust117.tnt1.yakima.wa.da.uu.net	7	95	1	86
sdsl-208-185-235-100.dsl.sjc.megapath.net	5	5	3	3
130.212.20.72	5	5	2	2
202.84.134.141	4	4	3	3
ubr-33.58.115.unionpark.cfl.rr.com	2	2	2	2
CBL187.pool010.CH001-riverside.dhcp.hs.earthlink.net	1	1	1	1
211.48.210.193	1	1	1	1
birx22ms1.teliamobile.net	1	1	1	1
211.50.30.241	1	1	1	1
w142.z208037228.nyc-ny.dsl.cnc.net	1	1	1	1
195.57.62.153	1	1	1	1
61.9.26.50	1	1	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.6.15	26	98	9	15
192.168.15.127	6	11	6	11
192.168.100.130	5	19	5	11
192.168.133.189	1	3	1	3
192.168.133.199	1	3	1	3
192.168.133.100	1	2	1	2
192.168.133.65	1	3	1	3
192.168.133.74	1	3	1	3
192.168.133.111	1	2	1	2
192.168.133.103	1	2	1	2
192.168.133.75	1	2	1	2
192.168.133.104	1	2	1	2
192.168.94.75	1	2	1	2
192.168.133.87	1	3	1	3
192.168.133.141	1	4	1	4
192.168.133.225	1	2	1	2

"An external host is attempting to use the RPC services of the target host." [CHL]

3.5.20. SITE EXEC – Possible wu-ftpd exploit & site exec-Possible wu-ftpd exploit

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
209.162.94.11	1	1	1	1
cm47580-a.ftwrth1.tx.home.com	1	1	1	1
adsl-64-217-116-106.dsl.hstntx.swbell.net	1	1	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.156.127	1	3	1	3
192.168.130.98	1	3	1	3
192.168.97.162	1	2	1	2

"There are several vulnerabilities in wu-ftpd. Wu-ftpd is a common package used for FTP services. For example, wu-ftpd, has a buffer overflow vulnerability due to improper bounds checking. For more information see http://www.cert.org/advisories/CA-1999-13.html." [CHL]

3.5.21. Tiny Fragments-Possible Hostile Activity

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
202.205.5.10	521	521	1	1
202.101.43.222	460	460	2	2
61.134.9.133	458	458	3	3
61.140.75.3	415	415	2	2
202.96.96.3	391	391	2	2
210.12.160.130	385	385	3	3
202.108.43.152	326	326	3	3
202.108.43.151	286	286	3	3
61.140.75.5	285	285	2	2
202.101.43.220	243	243	2	2
61.140.75.4	236	236	2	2
61.155.13.3	199	199	2	2
202.101.43.223	171	171	2	2
4.4.4.4	112	112	1	1
8.8.8.8	56	56	1	1
61.134.9.134	32	32	2	2
209.247.204.244	8	8	2	2
Destination-IPs	**# Alerts (sig)**	**# Alerts (total)**	**# Srcs (sig)**	**# Srcs (total)**
192.168.1.8	3148	3219	17	31
192.168.1.10	1264	1279	14	22
192.168.217.162	727	733	1	6
192.168.60.11	168	1457	2	140
192.168.1.9	8	16	5	10
208.162.62.208	7	7	1	1
192.168.202.18	6	10	1	5
192.168.100.230	5	808	3	15
192.168.98.123	2	51	1	7
192.168.215.106	2	6	1	5
192.168.201.14	1	6	1	6
192.168.219.46	1	4	1	4
192.168.71.38	1	4	1	4

"Usually the TCP header is 20 bytes long; but there are tools that can craft fragmented packets to bypass firewalls or Intrusion Detection Systems. Tiny packets can also be used for reconnaissance purposes." [CHL]

3.5.22. Happy 99 Virus

Source-IPs	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
ffml.fanfic.com	1	1	1	1

Destination-IPs	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
192.168.6.47	1	24	1	10

"The Happy 99 virus is usually attached to an email. When they execute the .exe file, there is a fireworks display to distract the target while the virus infects the target host." [CHL]

3.6. Summary

It is very obvious that the GIAC network is very interesting to potential attackers. I would recommend the following measures:

- totally block access from the Israelian ISDNNET and the Chinese NCFC / Chinese Academy of Sciences network. Hosts in that network play a very significant role as far as attacks are concerned. In fact, most attacks are coming from those networks. Unless absolutely necessary, any access should be blocked for machines originating from that network
- make sure that the latest patches are always installed, on all publicly accessible machines and internetworking devices (routers)
- block RPC access to machines from the outside network
- block access to tcp port 515 (the printer daemon) from the outside network
- restrict usage of Instant Messaging services as much as possible. ICQ is a very insecure protocol.
- Restrict usage of Napster and Gnutella services. These services are potential security risks and occupy precious bandwidth of the GIAC internet connection anyway.
- Block any ICMP echo requests on the border routers to the Internet.
- If not already happened, set up a firewall on the borders to the Internet and have it professionally maintained.
- Set up your mail servers with attachment scanning techniques, so that potentially dangerous Email attachments will be detected and eliminated.

3.7. Analysis process

I downloaded all the zip files (alert.zip, OOS.zip, scan1.zip, scan2.zip) and extracted them in separate directories. I also found (as other students before me) that the logfiles are terribly big (a concatenated allalerts.txt was about 53 MB) and that I had to use one my company's NT server with 512 MB RAM and 2 GB virtual memory in order to be able to run SnortSnarf and let it complete successfully. I though about playing around with SnortSnarf a little bit and trying to make it use less memory (up to 500 MB physical RAM while it was running), but time was too short, so I didn't.

I had to replace all the MY.NET. – entries in the alert files with valid IP address octets, I chose "192.168". This was necessary for SnortSnart to generate any useful output.

After it completed on the NT server, I wondered what to do next with it. "Import it into Excel, so you get these nice tables".. cool idea, the managers at GIAC like nice tables, so I did that.. before that, I had to clean up the HTML files created by SnortSnarf and re-format it into textfiles. I wrote a couple of VERY SIMPLE PERL scripts (do not judge me on those scripts, please, I just didn't have time to re-phrase them into "cooler" scripts before I submitted this ☺), in detail:

- index2html.pl, reformats and strips down the index.html that contains the overall summaries
- sig2html.pl, does the same to the sigXX.html files that contain the summaries per alert

To gather the Top 50 Attackers and the Top 50 Victims, I wrote more scripts, GatherAttackers.pl and GatherVictims.pl which go through the sigXX.txt files and concatenate the information in ASCII files that contain the number of alerts per IP. I then used the infamous

```
cat <textfiles> | sort –rn > results.txt
```

(respective names used for the textfiles and the results files) to get a sorted list, by descending order of all the IPs. I imported these files into Excel, cut out the Top 50 Victims and Attackers and pasted that into this document.

Tools used for analysis:

- SnortSnarf to generate HTML summaries of the alert-files. [SNORTSNARF]
- self written PERL scripts to convert the HTML pages into textfiles legible by MS Excel 2000

- cat, grep, sort for extracting additional information from the logfiles
- UnixDos for Windows 5.0 [UNIXDOS], to have at least some useful tools on Windows
- ActivePerl for Windows (to run SnortSnarf) [ACTIVEPERL]

Suggestions for the SnortSnarf guys:

- remove the usage of subroutines in the main part of the script, that should increase performance a little bit
- make it create several files while it is running, process them and have them merged together at the end, invisible to the user.
- include that option as command-line argument to generate output in ASCII/textfile format, tab- or comma separated
- somebody: rewrite it in C.

3.7.1. index2txt.pl

```
#!perl
open(HTML,"$ARGV[0]") || die ("cant open file $ARGV[0]\n");
@html=<HTML>;
close(HTML);

$txtfile=$ARGV[0];
$txtfile=~s/html/txt/;

while ( ( $html[$line]!~/Signature/ )  && ($line <= $#html) )
{
        $line++;
}
$line++;

open(TXT,">$txtfile") || die ("cant open $txtfile for writing.\n");

print TXT ("Signature\t# Alerts\t# Sources\t# Destinations\n");

while ( ($html[$line] !~ /\/TABLE/) && ($line <= $#html) )
{
        chomp($html[$line]);

        $html[$line]=~s/\///g;
        $html[$line]=~s/TR//g;
        $html[$line]=~s/<|>//g;
        $html[$line]=~s/TD/ /g;
        $html[$line]=~s/\!/ /g;
        $html[$line]=~s/(a href.*)//g;
        $html[$line]=~s/^\s+//g;

        print TXT ("$html[$line]\n");
        $line++;
}
```

3.7.2. GatherAttacker.pl

```
#!/usr/bin/perl

@dir=`dir -1 sig*`;

foreach $file (0..$#dir)
{
```

```
        if( ($filename)=($dir[$file]=~/ (sig.*txt)/) )
        {
                &calc($filename);
        }
}

sub calc {

my($line)=0;
my($attack)="";
my(@txt);


$txtfile=$_[0];
open(TXT,"$txtfile") || die ("cant open file $txtfile\n");
@txt=<TXT>;
close(TXT);

while ( ( $txt[$line]!~/Source.*Alerts/ )  && ($line <= $#txt) )
{
        $line++;
}
$line++;

while (( $line <= $#txt ) && ($txt[$line]!~/Destination/) )
{
        chomp($txt[$line]);
        if(($ip,$alerts)=$txt[$line]=~/^\s*(\d+\.\d+\.\d+\.\d+)\s+(\d+)\s+.*/)
        {
                print ("$alerts\t$ip\n");
        }
        $line++;
}

}
```

3.7.3. gatherVictims.pl

```
#!/usr/bin/perl

@dir=`dir -1 sig*`;

foreach $file (0..$#dir)
{
        if( ($filename)=($dir[$file]=~/ (sig.*txt)/) )
        {
                &calc($filename);
        }
}


sub calc {

my($line)=0;
my($attack)="";
my(@txt);


$txtfile=$_[0];
open(TXT,"$txtfile") || die ("cant open file $txtfile\n");
@txt=<TXT>;
close(TXT);

while ( ( $txt[$line]!~/Destination.*Alerts/ )  && ($line <= $#txt) )
{
        $line++;
```

```
}
$line++;

while ( $line <= $#txt )
{
        chomp($txt[$line]);
        ($ip,$alerts)=$txt[$line]=~/^\s*(\d+\.\d+\.\d+\.\d+)\s+(\d+)\s+.*/;
        print ("$alerts\t$ip\n");
        $line++;
}

}
```

3.7.4. sig2txt.pl

```
#!/usr/bin/perl
use Socket;

@dir=`dir -1 sig*`;

foreach $file (0..$#dir)
{
        if( ($filename)=($dir[$file]=~/ (sig.*html)/) )
        {
                &convert($filename);
                print("$filename...\n");
        }
}

sub convert {

my($line)=0;
my($attack)="";
my(@html);

$htmlfile=$_[0];
open(HTML,"$htmlfile") || die ("cant open file $htmlfile\n");
@html=<HTML>;
close(HTML);

$txtfile=$_[0];
$txtfile=~s/html/txt/;

while ($line<=$#html && $attack eq "")
{
        if($html[$line] =~ /et al for signature: (.*)<\/title>/)
        {
                $attack=$1;

        }
        $line++;
}


while ( ( $html[$line]!~/Source.*Alerts/ )  && ($line <= $#html) )
{
        $line++;
}
$line++;

## source IPs


open(TXT,">txt/$txtfile") || die ("cant open $txtfile for writing.\n");

print TXT ("Statistics for attack \"$attack\":\n");
```

```
print TXT ("Source\t# Alerts (sig)\t# Alerts (total)\t# Dsts (sig)\t# Dsts (total)\n");

while ( ($html[$line] !~ /\/TABLE/) && ($line <= $#html) )
{
        chomp($html[$line]);

        $html[$line]=~s/\/a/ /g;
        $html[$line]=~s/\///g;
        $html[$line]=~s/TR|tr//g;
        $html[$line]=~s/<|>//g;
        $html[$line]=~s/TD|td/ /g;
        $html[$line]=~s/\!/ /g;
        $html[$line]=~s/(a href.*)\"//g;
        $html[$line]=~s/^\s+//g;
        $host="";
        ($ipaddr,$rest)=$html[$line]=~/^(\d+\.\d+\.\d+\.\d+)(\s+.*)$/;

        $ip = inet_aton("$ipaddr");
        $host  = gethostbyaddr($ip, AF_INET);

        $host=$ipaddr if $host eq "";
        print TXT ("$host $rest\n");
        print ("$txtfile: $host $rest\n");
        $line++;
}

while ( ( $html[$line]!~/Destinations.*Alerts/ )  && ($line <= $#html) )
{
        $line++;
}
$line++;

print TXT ("Destination\t# Alerts (sig)\t# Alerts (total)\t# Srcs (sig)\t# Srcs
(total)\n");

while ( ($html[$line] !~ /\/TABLE/) && ($line <= $#html) )
{
        chomp($html[$line]);

        $html[$line]=~s/\/a/ /g;
        $html[$line]=~s/\///g;
        $html[$line]=~s/TR|tr//g;
        $html[$line]=~s/<|>//g;
        $html[$line]=~s/TD|td/ /g;
        $html[$line]=~s/\!/ /g;
        $html[$line]=~s/(a href.*)\"//g;
        $html[$line]=~s/^\s+//g;

        print TXT ("$html[$line]\n");
        $line++;
}

close(TXT);

}
```

4. References

In alphabetic order.

[ACTIVEPERL]
Perl for Windows from ActiveState.
http://www.activestate.com/Products/ActivePerl/

[BAKOS]
George Bakos' explanation of the rpc.statd / STATDX exploit.
http://www.sans.org/y2k/practical/George_Bakos.html#exploit

[CHL]
Chan Hee Lee's practical assignment.
http://www.sans.org/y2k/practical/Chan_Hee_Lee_GCIA.zip

[GCIA]
LevelTwo Intrusion Detection In Depth GCIA Practical Assignment
http://www.sans.org/giactc/GCIA_assign_27a.htm

[NEOHAPSIS]
Message about overflow bug in Sygate DNS server.
http://archives.neohapsis.com/archives/win2ksecadvice/2000-q2/0189.html

[RFC1858]
RFC 1858 - Security Considerations for IP Fragment Filtering.
http://www.faqs.org/rfcs/rfc1858.html

[SANSSNMP]
SANS ID FAQ: Using SNMP for Reconnaissance.
http://www.sans.org/newlook/resources/IDFAQ/SNMP.htm

[SANSRUSSIA]
SANS Reports about scans from a Russian domain.
http://www.sans.org/y2k/072818.htm

[SNORTPORT]
Port Search engine at Snort.org.
http://www.snort.org/Database/portsearch.asp

[SNORTARCHIVE]
The archive for the Snort Logfiles for this assignment.
http://www.sans.org/giactc/snort/index.htm

[STEARNS]
William Stearn's Practical Assignment.
http://www.sans.org/y2k/practical/william_stearns_gcia.html

[SPAN]
Cisco documentation regarding the switch SPAN port feature.
http://www.cisco.com/warp/public/473/41.html#intro

[UNIXDOS]
UNIX on DOS Toolkit.
http://www.profsoftware.com

[XFORCE-UNICODE]
IIS Unicode Bug explanation at XFORCE.
http://xforce.iss.net/alerts/advise68.php

[XFORCE-BO]
Back Orifice explanation at XFORCE.
http://xforce.iss.net/alerts/advise31.php

5. Credits

Chan Hee Lee for letting me borrow some of his attack description overviews without his knowledge ☺