GIAC Certified Intrusion Analyst (GCIA) Certification

Practical Assignment

Version 3.2

Bernard Kan
Oct 2002

Assignment 1 – Describe the State of Intrusion Detection: Spam Fighting by Technique of Intrusion Detection

Introduction

What is spam? Sometimes, the classification of whether a mail is a spam is quite subjective. Here is a definition of "spam" according to mail-abuse.org:

> An electronic message is "spam" IF: (1) the recipient's personal identity and context are irrelevant because the message is equally applicable to many other potential recipients; AND (2) the recipient has not verifiably granted deliberate, explicit, and still-revocable permission for it to be sent; AND (3) the transmission and reception of the message appears to the recipient to give a disproportionate benefit to the sender.

Spam mails has been a problem to Internet users recent years. A recent publish on ZDNet News stated that, 36% of e-mail traffic was spam. (Look here: http://zdnet.com.com/2100-1106-955842.html). It means out of every three mails you received, one was spam.

In my working environment (a local ISP), spam mail is a serious security issue. Not only our subscribers are disturbed by spam mails, spam mails also send out from our network too. Sometimes, spam mails are send out by our subscribers and sometimes spam mails are send out from loosely configured open relays of our business users. Many of those spam mails involve offensive sex contents. Occassionally, our network or mail servers were blocked by external parties because of spam mails. This is really a serious impact to our business as a ISP.

For the last 12 months, I analyzed varies kinds of spammer activities and found that techniques of Intrusion Detection can be deployed to mail servers and network IDS to monitor and minimize the activities of spammers. This article is a discussion of what I have found about spammers and what I have done to against them .

How Spammers Work

In order to send a mail, one must have two things: mail content and e-mail address of recipient. For a spammer, mail content is absolutely no problem at all. They have tons of content to send and many of them are promotional in nature. Promote some kind of pills, some kind services (include sex) or how to earn extra money. But how about recipients? What can spammers do if they want to send mails to millions of users on the Internet? It has been widely known that there are tools that can collect e-mail addresses from web sites and news group. But what if a spammer want to send spam mails to a particular group of users of a mail server or a ISP? One way is to perform a dictionary attack. The following were some logs that I have collected months ago showing how spammer do this.

```
20010912 133015420+0800 xxxm009dat mta 483 775 150 Warn;AcctUnknownUser(50/9)
user=intuition's@mydomain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
20010912 133015457+0800 xxxm009dat mta 483 720 693 Warn;AcctUnknownUser(50/9)
user=invader@mydomain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
20010912 133015484+0800 xxxm009dat mta 483 800 236 Warn;AcctUnknownUser(50/9)
user=invalid@mydomain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
20010912 133015529+0800 xxxm009dat mta 483 721 595 Warn;AcctUnknownUser(50/9)
user=invalidation@mydmain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y..231.19
20010912 133015542+0800 xxxm009dat mta 483 773 133 Warn;AcctUnknownUser(50/9)
user=invariant@mydomain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
20010912 133015557+0800 xxxm009dat mta 483 775 150 Warn;AcctUnknownUser(50/9)
user=invasion@mydomain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
20010912 133015589+0800 xxxm009dat mta 483 720 693 Warn;AcctUnknownUser(50/9)
user=invented@domain.com:from=<postmast@localhost.com>:fromhost=x.x.183.21:
localAddr=y.y.231.19
…snipped
```

The logs were collected from one of our mail servers. The spammer used a list of dictionary words as user names and tried to send it through. Of course, the successful rate was certainly low. Many users made up their user names with first name, last name and initials. Seldom did a user make up his user name with a dictionary word. However, when you look at the time stamp, you will notice that there were many attempts per second. The spammer tried so hard that it deteriorated the performance of our mail system.

Another kind of attempt would guarantee a much higher successful rate but it even caused us much more trouble. It’s a brute force attack. Look at the follow logs.

```
20010831 155928032+0800 xxxm016dat mta 3588 1584 631
Warn;AcctUnknownUser(50/9)
user=dfx4x@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928059+0800 xxxm016dat mta 3588 1508 891
Warn;AcctUnknownUser(50/9)
user=dfx4z@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928064+0800 xxxm016dat mta 3588 976 1047
Warn;AcctUnknownUser(50/9)
user=dfx4y@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928070+0800 xxxm016dat mta 3588 1633 377
Warn;AcctUnknownUser(50/9)
user=dfx4_@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928097+0800 xxxm016dat mta 3588 1640 1166
Warn;AcctUnknownUser(50/9)
user=dfx45@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928099+0800 xxxm016dat mta 3588 1506 1076
Warn;AcctUnknownUser(50/9)
user=dfx44@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
20010831 155928102+0800 xxxm016dat mta 3588 1530 1040
Warn;AcctUnknownUser(50/9)
user=dfx46@mydomain.com:from=<hiabc@ohdd.com>:fromhost=x.x.84.7:
localAddr=y.y.231.26
```

The spammer might be running some kind of scripts to go through the alphabets to make up possible user names. In our case, probably dfx4a,dfx4b,…dfx4z, ..dfx40, …dfx49.

As a large ISP, we have a database that stored all our user information and running at back end. This kind of brute force attack really affected the performance of our back end database system and thus affecting the overall performance of the mail system including sending and retrieving mails. Fortunately, we saw from the logs above that the problem was caused by a single IP address. Blocking that IP address at router immediately cooled the system down. However, the next one detected early this year, was a big problem.

The spammer used a open relay list to launch the brute force attack. Open relays are poorly configured mail servers on the Internet that send mails to anywhere whosever ask them. In the follow incident, we saw brute force attack from mail servers all over the world at the same time.

```
20020108 023834088+0800 xxxm009dat mta 4423 194 746 Warn;AcctUnknownUser(50/9)
user=aaaa8r@mydomain.com:from=<support@mail.com>:fromhost=212.68.199.80:
localAddr=x.x.231.19
20020108 023838658+0800 xxxm009dat mta 4423 200 1303 Warn;AcctUnknownUser(50/9)
user=aaaa85@mydomain.com:from=<support@mail.com>:fromhost=212.27.216.173:
localAddr=x.x.231.19
20020108 023838824+0800 xxxm009dat mta 4423 214 420 Warn;AcctUnknownUser(50/9)
user=aaaa9e@mydomain.com:from=<support@mail.com>:fromhost=212.205.88.29:
localAddr=x.x.231.19
20020108 023838905+0800 xxxm009dat mta 4423 194 746 Warn;AcctUnknownUser(50/9)
user=aaaa9m@mydomain.com:from=<support@mail.com>:fromhost=212.68.199.80:
localAddr=x.x.231.19
20020108 023838906+0800 xxxm009dat mta 4423 85 1208 Warn;AcctUnknownUser(50/9)
user=aaaa9i@mydomain.com:from=<support@mail.com>:fromhost=212.205.88.29:
localAddr=x.x.231.19
20020108 023839625+0800 xxxm009dat mta 4423 159 638 Warn;AcctUnknownUser(50/9)
user=aaaa9y@mydomain.com:from=<support@mail.com>:fromhost=212.186.85.200:
localAddr=x.x.231.19
20020108 023839776+0800 xxxm009dat mta 4423 62 34 Warn;AcctUnknownUser(50/9)
user=aaaa9g@mydomain.com:from=<support@mail.com>:fromhost=194.88.42.155:
localAddr=x.x.231.19
20020108 023839904+0800 xxxm009dat mta 4423 200 1303 Warn;AcctUnknownUser(50/9)
user=aaaa97@mydomain.com:from=<support@mail.com>:fromhost=212.27.216.173:
localAddr=x.x.231.19
…snipped
```

In the above incident, a spammer, claiming to be support@mail.com (which was fake) used a brute force attack and tried to send mails to our mail servers. Source IP addresses were believed to be open relays as those IP addresses were located physically at different part of the world. In this incident, we have observed over 800 different source IP addresses in this attack. No IP blocking could be performed at the router as the list was so large.

Besides attacking SMTP servers, we also observed attack on POP3 servers. Here was a example:

```
20011004 001208780+0800 xxxm013dat popserv 10354 2454 2578735
Note;AcctUnknownUser(50/9) user=wabcdefl@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 001208919+0800 xxxm013dat popserv 10354 2492 2578736
Note;AcctUnknownUser(50/9) user=wabcdefk@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 001209054+0800 xxxm013dat popserv 10354 2454 2578737
Note;AcctUnknownUser(50/9) user=wabcdefl@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 001209189+0800 xxxm013dat popserv 10354 2454 2578739
Note;AcctUnknownUser(50/9) user=wabcdefh@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 001209324+0800 xxxm013dat popserv 10354 2489 2578741
Note;AcctUnknownUser(50/9) user=wabcdefg@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 001209463+0800 xxxm013dat popserv 10354 2414 2578743
Note;AcctUnknownUser(50/9) user=wabcdefj@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 002708795+0800 xxxm013dat popserv 10354 2415 2583924
Note;AcctUnknownUser(50/9) user=wabcdefl@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
20011004 002708932+0800 xxxm013dat popserv 10354 2454 2583926
Note;AcctUnknownUser(50/9) user=wabcdefk@mydomain.com:cmd=PASS
<pswd>:fromhost=x.x.74.40
…snipped
```

The only reason for brute force attack on POP3 server was to enumerate a list of valid user names. However, this enumeration should not work on our POP3 server since no matter the user name exist or not, our POP3 server always return the same error message if authentication failed. I will have a elaboration on this later.

How We Defence - Against Brute Force Attack to SMTP Servers

The reader of this article may think that I were suffered badly. Yes. I have been waken up at night many times to handle all these situations. However, with the technique of Intrusion Detection, I quickly build up my defence mechanism.

According to rfc821, Simple Mail Transfer Protocol, SMTP server will return a 3 digits reply code during communication with mail client. For example:

```
bash$ telnet mail.mydomain.com 25
Trying x.x.48.136...
Connected to mail.mydomain.com.
Escape character is '^]'.
220 mail.mydomain.com ESMTP server (XXXMail vM.5.01.04.19 201-253-122-122-119-
20020516) ready Tue, 1 Oct 2002 22:28:53 +0800
helo test
250 mail.mydomain.com
mail from: kbernard@mydomain.com
250 Sender <kbernard@mydomain.com> Ok
rcpt to: no-such-user@mydomain.com
550 Invalid recipient: <no-such-user@mydomain.com>
quit
221 mail.mydomain.com ESMTP server closing connection
Connection closed by foreign host.
```

From the above illustration, we see that the server return a 550 code when I entered an invalid recipient. Here is a list of reply codes extracted from rfc821:

211 System status, or system help reply
214 Help message
220 <domain> Service ready
221 <domain> Service closing transmission channel
250 Requested mail action okay, completed
251 User not local; will forward to <forward-path>
354 Start mail input; end with <CRLF>.<CRLF>
421 <domain> Service not available, closing transmission channel
450 Requested mail action not taken: mailbox unavailable
451 Requested action aborted: local error in processing
452 Requested action not taken: insufficient system storage
500 Syntax error, command unrecognized
501 Syntax error in parameters or arguments
502 Command not implemented
503 Bad sequence of commands
504 Command parameter not implemented
550 Requested action not taken: mailbox unavailable
551 User not local; please try <forward-path>
552 Requested mail action aborted: exceeded storage allocation
553 Requested action not taken: mailbox name not allowed
554 Transaction failed

What I am most interested were some reply codes that may be caused by spamming activities, e.g. 500, 501, 550 and 553. What I have done was installed snort on all SMTP servers with the following rules:

```
alert tcp any 25 -> any any (msg: "SMTP 500 - Command unrecognized"; flags: A+;
content: 500";depth:3;nocase;)
alert tcp any 25 -> any any (msg: "SMTP 501 - Syntax error"; flags: A+;
content: "501"; depth:3;nocase;)
alert tcp any 25 -> any any (msg: "SMTP 550 - Mailbox unavailable"; flags: A+;
content: "550";depth:3;nocase;)
alert tcp any 25 -> any any (msg: "SMTP 553 - Mailbox-name not-allowed"; flags: A+;
content: "553";depth:3;nocase;)
```

Alerts were triggered whenever someone try to send mail to an invalid account. The following logs were triggered by spamming activities:

```
Oct  1 03:13:54 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
210.82.106.145:3395
Oct  1 03:13:54 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
61.142.97.97:3288
Oct  1 03:13:55 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
61.142.97.97:3288
Oct  1 03:13:56 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
213.42.143.227:4949
Oct  1 03:13:58 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
61.142.97.97:3288
Oct  1 03:13:59 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
61.142.97.97:3288
Oct  1 03:14:00 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
195.40.1.45:1307
Oct  1 03:14:00 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
61.142.97.97:3288
Oct  1 03:14:01 xxxp106 snort[295]: SMTP 550 - Mailbox unavailable: x.x.22.69:25 ->
200.204.196.9:3575
```

Note that in the above logs, our SMTP server x.x.22.69.25 reply a 550 code to the IP address at the rightmost column.

Now I have a lot of these alerts. But not all IP addresses that triggered these alerts are spammers. Ordinally users may misspell an e-mail address, or the recipient user has cancelled the account and changed to another ISP. What I did was to write a script to count the frequency of these IP addresses and extract those IP addresses that exceed a certain threshold.

```
#!/bin/sh
tail -500 /var/adm/messages | grep "SMTP 5" | awk '{print $13}' | awk -F: '{print $1}' | sort|
/usr/local/etc/count.awk | /usr/bin/awk '$1>25 {print $2}' >> /usr/local/etc/badlist.unsorted
```

The script above extract last 500 logs from /var/adm/messages (where snort stored the alerts) and locate those SMTP alerts, then extract those IP addresses at the rightmost column and then sort them. The results then passed to another script /usr/local/etc/count.awk to determine their frequency. If the frequency is greater than 25, then append the IP address to the file /usr/local/etc/badlist.unsorted. Here is the script count.awk.

```
$ cat count.awk
#!/usr/bin/awk -f
# This script count the frequency of first column (pre-sorted) and output
#   A
#   A          3  A
#   A     =>    1  C
#   C          2  B
#   B
#   B
NR == 1 {prev=$1;n=0}
$1 != prev {printf "%4d %s\n",n,prev;prev=$1;n=0}
$1 == prev {n++}
END {printf "%4d %s\n",n,prev}
```

I setup a cron job to run the first script every 5 minutes. Every 5 minutes, I'll have an updated list of spammer IP addresses in /usr/local/badlist.unsorted. Since our SMTP servers were Solaris system with ipfilter, I used the list to create a new ipfilter configuration file and reload it to block the spammers. Here is the script.

```
#!/bin/sh
cat /usr/local/etc/badlist.unsorted | sort -u | awk '{print "block in quick on hme0 proto tcp from
"$1" to any"}' > /etc/opt/ipf/ipf.conf
/etc/init.d/ipfboot reload
```

Of course, for users running other systems like Linux, similar functions can be implemented using ipchains or iptables. One simple method mentioned in http://spam.abuse.net was to add a simple host route for the spammer IP address:

```
route add -host <spammer IP> gw 127.0.0.1
```

Until then, I had an automatic defense system that can identify and block spammers dynamically. For the support@mail.com open relay brute force attack, I added the following rule to snort:

```
alert tcp any any -> INTERNAL 25 (msg: "Spam from support@mail.com"; flags: A+;
content: "support@mail.com";nocase;)
```

After adding the rule, following alerts were triggered.

```
Jan  2 17:07:10 xxxm009 snort[25615]: Spam from support@mail.com: 212.141.166.33:2574 ->
x.x.231.19:25
Jan  2 17:07:20 xxxm009 snort[25615]: Spam from support@mail.com: 212.129.240.82:24534 -
> x.x.231.19:25
Jan  2 17:07:21 xxxm009 snort[25615]: Spam from support@mail.com: 212.138.47.21:26082 ->
x.x.231.19:25
Jan  2 17:07:35 xxxm009 snort[25615]: Spam from support@mail.com: 212.130.48.54:4783 ->
x.x.231.19:25
Jan  2 17:07:38 xxxm009 snort[25615]: Spam from support@mail.com: 212.77.192.44:51809 ->
x.x.231.19:25
Jan  2 17:07:40 xxxm009 snort[25615]: Spam from support@mail.com: 212.131.128.130:4706 -
> x.x.231.19:25
Jan  2 17:07:44 xxxm009 snort[25615]: Spam from support@mail.com: 212.54.98.197:15759 ->
x.x.231.19:25
Jan  2 17:07:44 xxxm009 snort[25615]: Spam from support@mail.com: 212.131.128.130:4778 -
> x.x.231.19:25
Jan  2 17:07:45 xxxm009 snort[25615]: Spam from support@mail.com: 64.76.87.2:46280 ->
x.x.231.19:25
…snipped
```

Blocking these open relays IP addresses just need a few lines of script.

Against Brute Force Attack at POP Servers:

POP servers do not return codes like SMTP servers. In order to block brute force attack against POP servers, I manually connected to port 110 of POP servers and checked how it responded.

```
bash$ telnet pop.mydomain.com 110
Trying x.x.48.38...
Connected to pop.mydomain.com.
Escape character is '^]'.
+OK XXXMail POP3 server ready.
user no-such-user
+OK please send PASS command
pass no-such-pass
-ERR invalid user name or password.
no-such-command
-ERR Invalid command; valid commands: USER, QUIT
quit
```

From the above illustration, we see that whenever there was an authentication failed (either invalid user name or wrong password), the server replied "-ERR invalid user name or password." That's why enumeration of valid user names was not feasible here. The snort rules are simple:

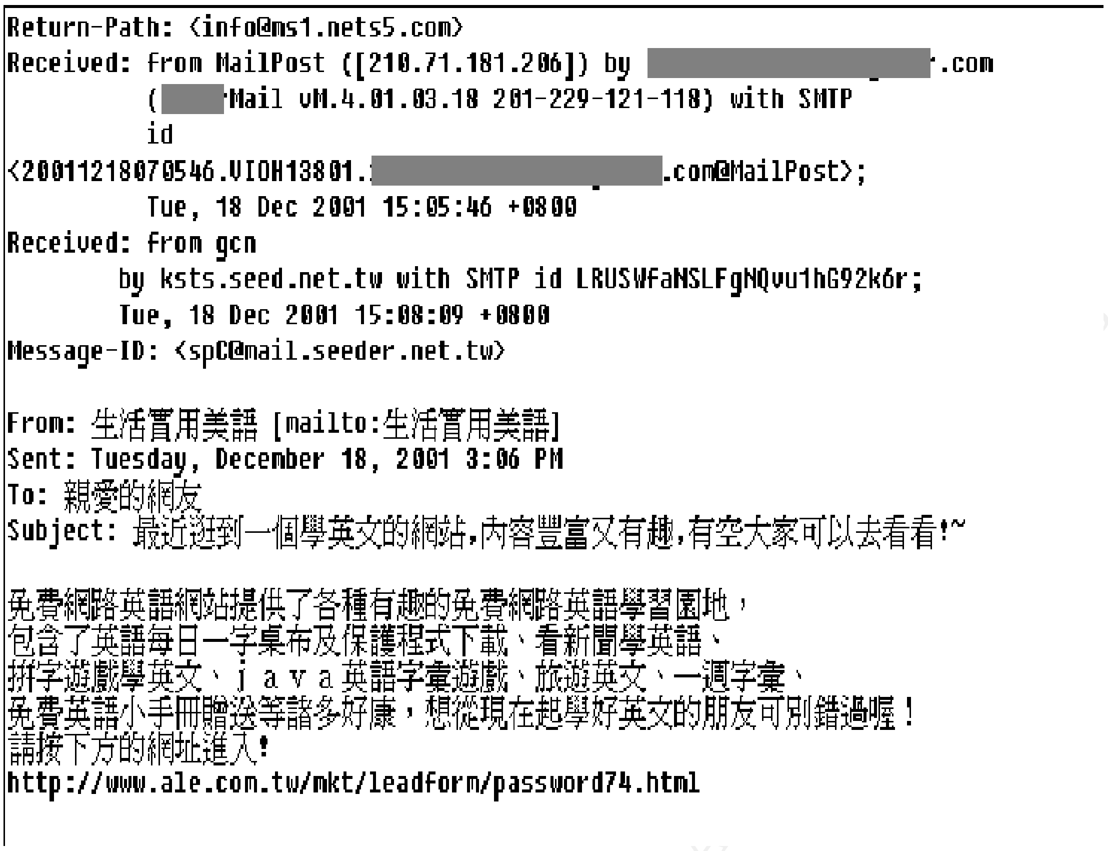
```
alert tcp any 110 -> any any (msg: "POP3 Error - Invalid command"; flags: A+;
content: "ERR Invalid command";)
alert tcp any 110 -> any any (msg: "POP3 Error - Invalid account"; flags: A+;
content: "ERR invalid user name or password";)
```

Against Internal Spammers and Open Relays

As mentioned in the Introduction, I worked in an ISP environment. Ocassionally, our subscribers also send out tons of spam mails. And almost everyday, there were loosely configured servers of business users used by spammers to send out bulk mails. The spam rules were then added to our network IDS. Thus we can monitor mail bomb activities of internal users to the Internet. The following logs, show a mail bomb against hotmail.com from one of our user. What we have done was blocked the internal IP address from sending mails to the Internet and then informed the user of the IP address to stop spamming activities. If the user refused to entertain our request, we have procedure to terminate his services. Most of the time users replied that their servers were used by spammers as relay.

```
Apr 24 15:52:06 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.135:25 -> x.x.64.198:1319
Apr 24 15:52:09 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.135:25 -> x.x.64.198:1319
Apr 24 15:52:23 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.7:25 -> x.x.64.198:1322
Apr 24 15:52:34 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.7:25 -> x.x.64.198:1322
Apr 24 15:52:37 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.7:25 -> x.x.64.198:1322
Apr 24 15:52:50 xxxids003.mydomain.com snort[16020]: SMTP 500 - Command
unrecognized: 64.4.49.7:25 -> x.x.64.198:1322
Apr 24 16:08:38 xxxids003.mydomain.com snort[16020]: SMTP 550 - Mailbox unavailable:
64.4.49.7:25 -> x.x.64.198:3787
Apr 24 16:20:35 xxxids003.mydomain.com snort[16020]: SMTP 550 - Mailbox unavailable:
64.4.56.199:25 -> x.x.64.198:3934
…snipped
```

The previous use of IDS based on the fact some spammers did not have an effective list of user names. When these spammers tried to send bulk mails, they triggered a lot of error codes from SMTP servers. However, in reality, there are skillful spammers that have an effective mailing list and they spammed so slowly that you won't notice their existence - until you received the spam yourself or you received a complaint from an angry user. Below was a spam example from Taiwan.



```
Return-Path: <info@ms1.nets5.com>
Received: from MailPost ([210.71.181.206]) by [redacted].com
          ([redacted]Mail vM.4.01.03.18 201-229-121-118) with SMTP
          id
<20011218070546.VIOH13801.[redacted].com@MailPost>;
          Tue, 18 Dec 2001 15:05:46 +0800
Received: from gcn
        by ksts.seed.net.tw with SMTP id LRUSWfaNSLFgNQvu1hG92k6r;
        Tue, 18 Dec 2001 15:08:09 +0800
Message-ID: <spC@mail.seeder.net.tw>

From: 生活實用美語 [mailto:生活實用美語]
Sent: Tuesday, December 18, 2001 3:06 PM
To: 親愛的網友
Subject: 最近逛到一個學英文的網站,內容豐富又有趣,有空大家可以去看看!~

免費網路英語網站提供了各種有趣的免費網路英語學習園地，
包含了英語每日一字桌布及保護程式下載、看新聞學英語、
拼字遊戲學英文、ｊａｖａ英語字彙遊戲、旅遊英文、一週字彙、
免費英語小手冊贈送等諸多好康，想從現在起學好英文的朋友可別錯過喔！
請按下方的網址進入!
http://www.ale.com.tw/mkt/leadform/password74.html
```

I have not sanitized the origin of the spam mail since I need those information for illustration. This Chinese spam mail was extracted from a user complain and it promoted an English teaching site http://www.ale.com.tw/mkt/leadform/password74.html.

Usually spam mail contains some HTTP links for promotion. The link can be treated as a characteristic or a "signature" of the spam mail. What I did was extracted the source IP address of the spam mail (in this case, 210.71.181.206) and removed the last octet and made it a class C network, i.e. 210.71.181.0/24. Then a portion of the HTTP was selected as the "signature" and put it in a snort rule:

```
alert tcp 210.71.181.0/24 any -> any any (msg: "Spam rule #40";flags: A+;content:
"www.ale.com.tw/mkt/le";)
```

A number was given to the rule for tracking. After I have collected enough spam mails (or complaints), a snort rule file for detecting spam mails was created.

```
alert tcp 128.121.16.0/24 any -> any any (msg: "Spam rule #173094";flags: A+;content:
"www.amazingofferings.com";)
alert tcp 128.121.16.0/24 any -> any any (msg: "Spam rule #202899";flags: A+;content:
"216.177.60.55";)
alert tcp 128.121.8.0/24 any -> any any (msg: "Spam rule #202180";flags: A+;content:
"bye.emf3.com";)
alert tcp 128.242.104.0/24 any -> any any (msg: "Spam rule #185805";flags: A+;content:
"www.webshots.com";)
alert tcp 129.217.131.0/24 any -> any any (msg: "Spam rule #197270";flags: A+;content:
"euroconnexions.com";)
alert tcp 130.244.199.0/24 any -> any any (msg: "Spam rule #187475";flags: A+;content:
"www.porn-dvd-movies.com@209.203.1";)
alert tcp 139.223.100.0/24 any -> any any (msg: "Spam rule #190949";flags: A+;content:
"quote.morningstar.com.";)
alert tcp 140.239.165.0/24 any -> any any (msg: "Spam rule #186492";flags: A+;content:
"relay.netatlantic.com";)
alert tcp 143.89.153.0/24 any -> any any (msg: "Spam rule #202359";flags: A+;content:
"UGetScholarships@vmadmin.com";)
alert tcp 144.134.163.0/24 any -> any any (msg: "Spam rule #206639";flags: A+;content:
"evc.to/nicegi";)
alert tcp 144.135.25.0/24 any -> any any (msg: "Spam rule #202697";flags: A+;content:
"209.81.54.22";)
…snipped
```

Why use a class C for source IP addresses? This simply increase the chance for a hit since the spammer may be using a dynamically allocated dialup IP address range.
When snort was activated on SMTP server with the rule file, spam mails were detected in syslog. Scripts can be used to block identified spammer IP addresses.

```
Jan  6 07:35:23 xxxm009 snort[25615]: Spam rule #41: 202.204.105.2:33116 -> x.x.231.19:25
Jan  6 07:36:19 xxxm009 snort[25615]: Spam rule #41: 202.204.105.2:33269 -> x.x.231.19:25
Jan  6 07:37:55 xxxm009 snort[25615]: Spam rule #56: 202.108.44.222:39268 ->
x.x.231.19:25
Jan  6 07:38:18 xxxm009 snort[25615]: Spam rule #56: 202.108.44.222:39527 ->
x.x.231.19:25
Jan  6 07:38:32 xxxm009 snort[25615]: Spam rule #56: 202.108.44.222:39674 ->
x.x.231.19:25
Jan  6 07:40:39 xxxm009 snort[25615]: Spam rule #68: 209.90.161.66:1619 -> x.x.231.19:25
Jan  6 07:41:12 xxxm009 snort[25615]: Spam rule #68: 209.90.161.55:1250 -> x.x.231.19:25
Jan  6 08:41:07 xxxm009 snort[25615]: Spam rule #96: 195.166.233.45:1334 -> x.x.231.19:25
Jan  6 10:21:08 xxxm009 snort[25615]: Spam rule #68: 209.90.161.121:1274 -> x.x.231.19:25
Jan  6 10:41:44 xxxm009 snort[25615]: Spam rule #68: 209.90.161.51:2580 -> x.x.231.19:25
Jan  6 10:54:32 xxxm009 snort[25615]: Spam rule #68: 209.90.161.120:2238 -> x.x.231.19:25
…snipped
```

Further Enhancement

Spammers changed their spam content frequently. In order to follow with spammers, I made the following automation to snort on SMTP servers:

- Spammer source IP addresses and signatures were extracted by scripts from a dummy mailbox and a mailbox that received complaints from users
- Snort rules were generated automatically from extracted source network information and signatures.
- New snort rules were automatically added to SMTP servers everyday

Conclusion

With technique of Intrusion Detection, I were able to set up a dynamic defense mechanism against spam mails with minimum human intervention. However, this defense mechanism still had short comings. Since cron jobs were used to scan the syslog to determine spammer IP addresses, there were "gaps" between cron jobs intervals that spammer IP addresses were allowed to send mails before blocking can take place. Some professional spammers who used a very long list of IP addresses were able to send thousands of spam mails between these gaps everyday on each server. Another problem was performance issue. When the list of blocked IP addresses grew longer and longer, system performance was affected eventually. IP addresses had to be released strategically to release system resources but meanwhile had to maintain the level of spam protection.

Reference

The IETF Anti Spam Recommendations URL: ftp://ftp.isi.edu/in-notes/bcp/bcp30.txt (18 Oct 2002)

Fight Spam on the Internet! (from abuse.net) URL: http://spam.abuse.net/ (18 Oct 2002)

Snort.org. Snort Users Manual. URL: http://www.snort.org/docs/writing_rules/ (18 Oct 2002)

RFC 2635 - why spam is evil. URL: http://sunsite.dk/RFC/rfc/rfc2635.html (18 Oct 2002)

RFC 821 - Simple Mail Transfer Protocol (SMTP). URL: http://www.ietf.org/rfc/rfc0821.txt (18 Oct 2002)

What is "spam"? URL: http://mail-abuse.org/standard.html (18 Oct 2002)

Robert Lemos. Spam hits 36 percent of e-mail traffic. URL: http://zdnet.com.com/2100-1106-955842.html (18 Oct 2002)

Assignment 2 – Network Detects

Detect #1 - DoS ATH0 Modem Disconnect

```
Oct  4 05:31:43 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:31:44 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:31:46 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:31:59 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:32:00 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:32:01 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:32:06 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:32:13 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224
Oct  4 05:32:24 xxxids004 snort[7823]: IDS264/dos-ath0-modem-disconnect:
x.x.152.229 -> 129.168.17.224

[**] IDS264/dos-ath0-modem-disconnect [**]
10/04-05:27:40.768904 x.x.152.229 -> 129.168.17.224
ICMP TTL:125 TOS:0x0 ID:565 IpLen:20 DgmLen:300
Type:8  Code:0  ID:0   Seq:0  ECHO
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
2B 2B 2B 41 54 48 30 0D 0A 00 00 00 00 00 00 00  +++ATH0.........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

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

[**] IDS264/dos-ath0-modem-disconnect [**]
10/04-05:27:40.784674 x.x.152.229 -> 129.168.17.224
ICMP TTL:125 TOS:0x0 ID:568 IpLen:20 DgmLen:300
Type:8  Code:0  ID:0   Seq:0  ECHO
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
2B 2B 2B 41 54 48 30 0D 0A 00 00 00 00 00 00 00  +++ATH0.........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
[**] IDS264/dos-ath0-modem-disconnect [**]
10/04-05:27:40.986485 x.x.152.229 -> 129.168.17.224
ICMP TTL:125 TOS:0x0 ID:610 IpLen:20 DgmLen:300
Type:8  Code:0  ID:0   Seq:0  ECHO
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
2B 2B 2B 41 54 48 30 0D 0A 00 00 00 00 00 00 00  +++ATH0.........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

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

1. Source of trace

The detect came from an IDS at the backbone of an ISP network. The source IP address was a dynamic IP of a broadband subscriber. It means that the detection of this attack was from our internal network to the external Internet.

2. Detect was generated by:

Snort 1.7 on a Solaris machine with Arachnids signature database. The alerts were generated by the follow rule:

```
alert ICMP $EXTERNAL any -> $INTERNAL any (msg: "IDS264/dos-ath0-
modem-disconnect"; itype: 8; content: "+++ath0"; nocase;)
```

In the snort configuration file, both $EXTERNAL and $INTERNAL were actually equal to "any" since we want to detect both incoming and outgoing hacking activities.

3. Probability the source address was spoofed

The attack was done by sending ICMP packets to the victim and actually did not need a reply. In general, the probability of spoofed source address was high. However, as the attack was detected in outgoing direction of ISP's backbone network and the IP address was a valid dynamic IP address used by subscriber, it was likely that the source IP address was not spoofed.

4. Description of attack

This event indicates that an attacker has attempted to force a modem to disconnect. An attacker has sent a special string in an ICMP ping packet. When the modem-connected system replies to the ping, the modem recognizes the reply traffic as a hang-up command. The victim is then disconnected. It is possible to knock a modem offline with +++ath0. The +++ puts the modem into command mode, and the ath0 hangs up the line.

5. Attack mechanism

Some time ago, Hayes Microcomputer Products got a patent -- known as the "Heatherington patent" -- on its method of doing modem escape sequences. The patent involved the timing of the escape sequence: The characters "+++" followed by a 1-second pause. To get around the patent, some modem vendors simply eliminated the pause, so that the sequence +++AT would bring the modem back to command mode in all cases. So, by forcing the victim to respond with the string "+++ATH0" many brands of modems will interpret the +++ATH0 as the user manually attempting to enter command mode and execute a command. Because of this, when the victim attempts to respond with the +++ATH0 the modem sees it within the IP datagram and hangs up the modem.

Today, modem chip vendors had already licensed the patent, so those modem manufacturers didn't need to mess with the patent issue anymore. Now, it's rare to find a modem that responds to the attack unless there happens to be a long pause in the data stream after the "+++".

However, many old modems on the Internet are still susceptible to this vulnerability.

Max Schau (Noc-Wage, wage@IDIRECT.CA) mentioned some kind of "modem ping killer" by MrPhoenix (phoenix@iname.com) in a bugtraq posting on Sep 1998. (This point thanks to Chris Baker). To exploit a victim with such kind of old modems, you can either make your own program to send the required packet, or use the ping program with the *wonderful* option "-p" with which you can specify up to 16 bytes to fill out the packet to send. The "-p" option requires the pattern to be entered in hex digits. The equivalent of the '+++ATH0' string in hex is: 2b2b2b415448300d .

The complete command is : ping -p 2b2b2b415448300d victim_ip

NOTE: The "-p" option is not supported by the"ping" program from Microsoft shipped with MS-Windows.

In our detect above, the payload size was constantly 272 bytes and the command string "+++ATH0" consistently appeared at offset 32 of the payload. These packets seemed to be generated by tools other than ping command.

6. Correlations:

This was an old day DoS attack. I have searched Google for correlated incidents but seemed no one report such kind of attack in live.

However, in a posting by Michael H. Warfield (mhw@iss.net) in http://archives.neohapsis.com/archives/iss/2001-q2/0155.html, we found some interesting history. During those patent wars, the hayes employee's would put +++ATH0 on a new line in all of their usenet postings. As told by the author: "Needless to say, this caused widespread random mayhem and didn't enhance the reputation of either Hayes nor the modem manufactures one bit."

The author of the post also mentioned another incident.

"I recently (about 6 months ago) had an incident with some chumps "TIES bombing" an entire ISP. They were flood pinging his netblock with ICMP echo packets containing "\r+ + +ATH0\r" in the payload. What would happen was that, when a customer would connect in and get an IP address, the first ping would cause the ISP's (the outbound) modem to jump into command mode and then hang up the phone (they could create even MORE mayhem by issuing dial commands in the payload - think about it). The ISP was at a total loss trying to figure out why all his PPP dialins where hanging up within seconds of connecting till I suggested looking at this old problem. That was it. All of their banks of modems were TIES modems. They had to set the escape character to 255 or 127 to disable the escape. But that only fixed SOME of the connections. Some customers were still broken! The answer was simple! The echo packet was getting through and being echoed back. Then the customer's modem (transmitting back to the ISP) would see the + + +ATH0 and jump into command mode and hang up the phone. Both ends of the link had to be fixed to prevent TIES bombing." (Points in correlations here thanks to Chris Baker)

7. Evidence of active targeting:

Unknown. Since this attack only works on old models of modem, one interesting question is: Was the target 129.168.17.224 actively selected? I have searched our IDS logs but did not find any other alerts associated with source x.x.152.229 and target 129.168.17.224. A searched on WHOIS returned a record that 129.168.17.224 belongs to NASA (National Aeronautics and Space Administration) of US government. The reverse lookup of 129.168.17.224 failed and I could not determine whether it is a modem dialup IP address. So, this is still a mystery. May be this is just a wrong number that the target should be 192.168.17.224? As the IDS did not detect replies from remote IP address (the IDS should be able to detect them if any, as the string '+++ATH0' should be included in the reply packets as well), it was not clear whether the remote IP address was not actually online at that time or it got disconnected by the attack. However, the trace packets also have a carriage return linefeed after '+++ATH0' which makes the possibility of random data even less likely. (This point thanks to Bill Royds).

8. Severity:

Target Criticality: 4. The target may be a government desktop.

Attack Lethality: 4. The attack was a DoS.

System Countermeasures: 3. We have no information of target system. It may be vulnerable or not vulnerable.

Network Countermeasures: 2. We have no information of the target network.
We cannot block ICMP traffic in our ISP network but we had IDS that detected the attack.

Attack Severity = 4+4-3-2 = 3 (The calculation here thanks to Chris Baker)

9. Defensive recommendation:

1. If target system was running old models of modem, upgrade modem.
2. Put in modem initialization string "s2=255" which will disable the modem's ability to go into command mode. (Can cause problems for some people)
3. Block ICMP traffic at target network.
4. Use other way to connect to Internet instead of modem
5. Contact NASA and inform them of the attack and the defensive action taken against the attacker on our network.

10. Multiple Choice:

Which of the follow commands can caused a modem to disconnect:
A. +++ATM0
B. +++ATH0
C. ATM0+++
D. ATH0+++

Answer: B, "+++" put the modem to command mode, "ATH0" hang up the modem

Detect #2 - Tear Drop Attack

```
Oct 17 12:25:40 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification
: Misc activity] [Priority: 3]: {ICMP} x.x.23.50 -> 202.66.147.158
Oct 17 12:25:59 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification
: Misc activity] [Priority: 3]: {ICMP} x.x.23.50 -> 202.66.147.158
Oct 17 12:26:05 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification
: Misc activity] [Priority: 3]: {ICMP} x.x.23.50 -> 202.66.147.158
Oct 17 12:26:57 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification
: Misc activity] [Priority: 3]: {ICMP} x.x.23.50 -> 202.66.147.158
Oct 17 12:27:23 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification
: Misc activity] [Priority: 3]: {ICMP} x.x.23.50 -> 202.66.147.158

snipped…

Oct 17 14:09:28 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification: Misc activity] [Priority: 3]: {ICMP}
x.x.23.50 -> 202.66.147.158
Oct 17 14:09:29 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification: Misc activity] [Priority: 3]: {ICMP}
x.x.23.50 -> 202.66.147.158
Oct 17 14:09:59 xxxids002.mydomain.com snort[2298]: [1:1322:4] BAD TRAFFIC
bad frag bits [Classification: Misc activity] [Priority: 3]: {ICMP}
x.x.23.50 -> 202.66.147.158


[**] BAD TRAFFIC bad frag bits [**]
10/17-12:25:40.481650 x.x.23.50 -> 202.66.147.158
ICMP TTL:254 TOS:0x0 ID:31212 IpLen:20 DgmLen:1500 DF MF
Frag Offset: 0x0   Frag Size: 0x5C8
00 00 2E D8 42 F1 5C 05 61 62 63 64 65 66 67 68  ....B.\.abcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
```

```
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77                          pqrstuvw

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

[**] BAD TRAFFIC bad frag bits [**]
10/17-12:25:40.481764 x.x.23.50 -> 202.66.147.158
ICMP TTL:254 TOS:0x0 ID:31212 IpLen:20 DgmLen:1500 DF MF
Frag Offset: 0xB9   Frag Size: 0x5C8
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
```

```
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68   pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68   pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68   pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
```

```
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68                          abcdefgh

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

[**] BAD TRAFFIC bad frag bits [**]
10/17-12:25:59.466703 x.x.23.50 -> 202.66.147.158
ICMP TTL:254 TOS:0x0 ID:31244 IpLen:20 DgmLen:1500 DF MF
Frag Offset: 0xDBB   Frag Size: 0x5C8
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
```

```
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E                          ghijklmn

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

[**] BAD TRAFFIC bad frag bits [**]
10/17-12:25:59.466874 x.x.23.50 -> 202.66.147.158
ICMP TTL:254 TOS:0x0 ID:31244 IpLen:20 DgmLen:1500 DF MF
Frag Offset: 0x109F   Frag Size: 0x5C8
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61  ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71  bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A  rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63  klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73  defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C  tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65  mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75  fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E  vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67  opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77  hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62  jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72  cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B  stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64  lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74  efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D  uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66  nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76  ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F  wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68  pqrstuvwabcdefgh
```

```
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68   pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68   pqrstuvwabcdefgh
69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61   ijklmnopqrstuvwa
62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71   bcdefghijklmnopq
72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A   rstuvwabcdefghij
6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63   klmnopqrstuvwabc
64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73   defghijklmnopqrs
74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C   tuvwabcdefghijkl
6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65   mnopqrstuvwabcde
66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75   fghijklmnopqrstu
76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E   vwabcdefghijklmn
6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67   opqrstuvwabcdefg
68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77   hijklmnopqrstuvw
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70   abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62   jklmnopqrstuvwab
63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72   cdefghijklmnopqr
73 74 75 76 77 61 62 63 64 65 66 67 68 69 6A 6B   stuvwabcdefghijk
6C 6D 6E 6F 70 71 72 73 74 75 76 77 61 62 63 64   lmnopqrstuvwabcd
65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74   efghijklmnopqrst
75 76 77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D   uvwabcdefghijklm
6E 6F 70 71 72 73 74 75 76 77 61 62 63 64 65 66   nopqrstuvwabcdef
67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76   ghijklmnopqrstuv
77 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F   wabcdefghijklmno
70 71 72 73 74 75 76 77                           pqrstuvw
```

1. Source of trace

The detect came from an IDS at the backbone of an ISP network. The source IP address was a dynamic IP of a broadband subscriber. It means that the detection of this attack was from our internal network to the external Internet.

2. Detect was generated by:

Snort 1.8.1 on a Solaris machine with up-to-date signatures from snort.org. The alerts were generated by the follow rule:

```
alert ip $EXTERNAL_NET any -> $HOME_NET any
(msg:"BAD TRAFFIC bad frag bits"; fragbits:MD; sid:1322;
classtype:misc-activity; rev:4;)
```

In the snort configuration file, both $EXTERNAL_NET and $HOME_NET were actually equal to "any" since we want to detect both incoming and outgoing hacking activities.

3. Probability the source address was spoofed

The attack was done by sending ICMP packets to the victim and actually did not need a reply. In general, the probability of spoofed source address was high. However, as the attack was detected in outgoing direction of ISP's backbone network and the IP address was a valid dynamic IP address used by subscriber, it was likely that the source IP address was not spoofed.

4. Description of attack

This was likely a denial of service attack. The attacker send packets with both 'More Fragments' and 'Don't Fragment' flags ON at the same. This was something what we called 'Out of Specification'. This was a kind of abnormal traffic. Furthermore, the fragments seemed cannot put together to form a valid data segment. The overlapped by looking at the offsets. The was a kind of Teardrop attack. The attack also flooded the network with large ICMP packets and lasted over 100 minutes.

5. Attack mechanism

Fragmentation occurs when an IP datagram travelling on a network has to traverse a network with a maximum unit (MTU) that is smaller than the size of the datagram. If a datagram is larger than the MTU of the target network (say, 1500 bytes in Ethernet) it requires fragmentation by a router directing it to the target network. Fragmentation can also occur when a host needs to put a datagram on the network that exceeds the MTU. When fragmentation occurred, all fragmented packets except the last one, must have a flag called "More Fragments" (MF) turned on in the IP header. The offset values in the IP header of the fragments must enable the fragments to assemble back the original unfragmented packet. Inside the IP header, there is another flag called

"Don't Fragment" (DF) which is used to indicate that a packet should not be fragmented. These two flags are mutual exclusive - they cannot be ON at the same time.

For a detailed description of Fragmentation, readers can refer to the book "Network Intrusion Detection . An Analyst's Handbook" published by New Riders. This book also was a reference book for GCIA.

In the alerts above, the payload of the packets indicated that both MF and DF were ON. Windows or NT systems without up-to-dated patches may crashed when handling this traffic.

Now, let's look at the offset issue. The offset of the first packet payload was 0 and length of fragment was 1480 bytes (0x5c8). It had an ID 31212. The second packet also had the same ID 31212. (This point thanks to Oliver Viittamaki) However, the offset of the second packet payload was only 185 (0xB9), the fragment thus overlapped with the first one. I only quoted a few samples due to limitation of space here. All other packets had similar characeristics like those I quoted here: two or three packets with the same ID, but the offset values were so bad that the fragments could not assemble back together. This was called "Teardrop Attack" and able to crash some old systems that were not properly patched.

In most systems, the packet assembling process consumes CPU and memory. If incompleted fragments were received continously, the system would eventually exhause system resources. The system either remained unresponsive for a period of time (DOS) or simply crashed. (This point thanks to Oliver Viittamaki). If a vulnerable Windows system was attacked, it would show a blue screen of death.

The attacker's IP address was from an ADSL broadband network with down-stream 1.5M and up-stream 256K. The above attack seemed so rigorous that it used up all its up-stream bandwidth to perform the attack. Any site with bandwidth 256K or below simply no longer able to provide external services under this degree of flooding. However, when I tried to perform a reverse lookup of the target IP address, it yielded ip-158-147-66-202.rev.dyxnet.com, which seemed to be a kind of dynamic IP address provided by ISP dyxnet.com. It did not respond to ping at the time of writing this analysis.

6. Correlations:

I have searched Google for correlated incidents but seemed not much reported in live. However, there was an analysis subscribed by Scott Gregory <gregory@lurhq.com> to intrusions@incidents.org. Look here: http://cert.uni-stuttgart.de/archive/intrusions/2002/08/msg00106.html

The same snort alerts were reported in Scott's analysis however the alerts were triggered by CodeRed activities.

7. Evidence of active targeting:

Likely. The attack's IP address did not triggered other alerts of the IDS.
The attack was rigorous and last for over 100 minutes.

8. Severity:

Target Criticality: 3. The target seemed to be a client machine with
dynamic IP address.
Attack Lethality: 4. The affected system may crash if vulnerable to attack.
System Countermeasures: unknown (1 or above)
Network Countermeasures: 1. We have no information of the target network.
But we cannot block ICMP traffic in our ISP network.

Attack Severity = 3+4-unknown-1 = 6-unknown (on high side)

9. Defensive recommendation:

1. Update system with the most updated patches.

2. Block ICMP traffic at target network. However, this may cause difficulties in network operations as ICMP is used for path MTU determination. (This point thanks to Bill Royds). May be a client side personal firewall is a better solution.

3. Handle the attacker's account according to the ISP's Acceptable
Use Policy.

10. Multiple Choice:
Which combination of flags in TCP/IP is "Out-of-Specification"?
A. ACK-PUSH
B. ACK-MF
C. PUSH-MF
D. DF-MF

Answer: D. DF (Don't Fragment) and MF (More Fragments) are mutual exclusive

Detect #3 - Virus - Possible scr Worm

1. Source of trace

The data files came from http://www.incidents.org/logs/Raw/2002.6.11 and http://www.incidents.org/logs/Raw/2002.6.12. It is a little strange that the file name is "6.11" and "6.12" but contained data of "7/11" and "7/13". However I see a "correlation" of this phenomenon in other candidates' works as well. So I assume I am not doing anything wrong with the logs.

The network under monitoring here, 46.5.x.x (probably sanitized) seems not a very open network. Not much hacking activities detected by the IDS except a few port scan activities. An interesting thing here was that, 46.5.180.250 seemed to be a firewall that performing NAT and hiding architecture of internal network. I have chosen an alert that involved activities of this IP address.

2. Detect was generated by:

Snort-1.8.7-win32 with standard rule set but enables all the remarked options in the default snort.conf file.

I ran the command like this:
snort -c snort.conf -d -e -l log -r 2002.06.11

I find an interesting alert in 07/11.

```
[**] [1:729:3] Virus - Possible scr Worm [**]
[Classification: Misc activity] [Priority: 3]
07/11/02-19:18:34.754488 167.206.112.6:110 -> 46.5.180.250:63879
TCP TTL:58 TOS:0x0 ID:32260 IpLen:20 DgmLen:1500
***AP*** Seq: 0xCF099DAB  Ack: 0xAF11A  Win: 0xFAF0  TcpLen: 20
```

The I search other logs and find two similar alerts in 07/13.

```
[**] [1:729:3] Virus - Possible scr Worm [**]
[Classification: Misc activity] [Priority: 3]
07/13/02-02:23:57.684488 167.206.112.6:110 -> 46.5.180.250:61658
TCP TTL:58 TOS:0x0 ID:40103 IpLen:20 DgmLen:1500
***A**** Seq: 0x41CC58D9  Ack: 0xAE02C162  Win: 0xFAF0  TcpLen: 20

[**] [1:729:3] Virus - Possible scr Worm [**]
[Classification: Misc activity] [Priority: 3]
07/13/02-02:23:57.694488 167.206.112.6:110 -> 46.5.180.250:61658
TCP TTL:58 TOS:0x0 ID:40104 IpLen:20 DgmLen:1365
***AP*** Seq: 0x41CC5E8D  Ack: 0xAE02C162  Win: 0xFAF0  TcpLen: 20
```

The source IP 167.206.112.6 was probably a pop3 server (using port 110). When an internal host behind the firewall 46.5.180.250 retrieved some mails, it triggered the IDS alert.

Let's look at the rule that generated this alert.

```
alert tcp any 110 -> any any (msg:"Virus - Possible scr Worm";
content: ".scr"; nocase; sid:729;  classtype:misc-activity; rev:3;)
```

Well, the rule simply watch for the four characters ".scr" coming from pop server. ".scr" is an extension of screen saver in Microsoft Windows system. Many worms hide its true identity and "spoof" itself as a legitimate executable program (this time, a screen saver) and pass as an attachment in e-mails.

Since this rule was so simple, there was a high chance that the alerts were false positives. We have to look at the payload of the packets to determine this. Here is the payload of the packets of the corresponding alerts.

```
[**] Virus - Possible scr Worm [**]
07/11-19:18:34.754488 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x5EA
167.206.112.6:110 -> 46.5.180.250:63879 TCP TTL:58 TOS:0x0 ID:32260 IpLen:20
DgmLen:1500
***AP*** Seq: 0xCF099DAB  Ack: 0xAF11A  Win: 0xFAF0  TcpLen: 20
7A 6B 56 79 35 73 33 41 49 49 78 68 2B 6B 56 57  zkVy5s3AIIxh+kVW
67 29 0D 0A 43 6F 6E 74 65 6E 74 2D 69 64 3A 20  g)..Content-id: 
3C 49 68 34 32 34 37 58 57 31 54 55 77 69 3E 0D  <Ih4247XW1TUwi>.
0A 43 6F 6E 74 65 6E 74 2D 74 79 70 65 3A 20 61  .Content-type: a
75 64 69 6F 2F 78 2D 77 61 76 3B 20 6E 61 6D 65  udio/x-wav; name
3D 58 5A 58 39 30 30 41 2E 73 63 72 0D 0A 43 6F  =XZX900A.scr..Co
6E 74 65 6E 74 2D 74 72 61 6E 73 66 65 72 2D 65  ntent-transfer-e
6E 63 6F 64 69 6E 67 3A 20 62 61 73 65 36 34 0D  ncoding: base64.
0A 43 6F 6E 74 65 6E 74 2D 64 69 73 70 6F 73 69  .Content-disposi
74 69 6F 6E 3A 20 61 74 74 61 63 68 6D 65 6E 74  tion: attachment
3B 20 66 69 6C 65 6E 61 6D 65 3D 58 5A 58 39 30  ; filename=XZX90
30 41 2E 73 63 72 0D 0A 0D 0A 54 56 71 51 41 41  0A.scr....TVqQAA
4D 41 41 41 41 45 41 41 41 41 2F 2F 38 41 41 4C  MAAAAEAAAA//8AAL
67 41 41 41 41 41 41 41 41 41 51 41 41 41 41 41  gAAAAAAAAAQAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 0D 0A 41 41 41 41 41 41 41 41 41 32 41 41 41  AA..AAAAAAAAA2AAA
41 41 34 66 75 67 34 41 74 41 6E 4E 49 62 67 42  AA4fug4AtAnNIbgB
54 4D 30 68 56 47 68 70 63 79 42 77 63 6D 39 6E  TM0hVGhpcyBwcm9n
63 6D 46 74 49 47 4E 68 62 6D 35 76 64 43 42 69  cmFtIGNhbm5vdCBi
5A 53 42 79 64 57 34 67 61 57 34 67 0D 0A 52 45  ZSBydW4gaW4g..RE
39 54 49 47 31 76 5A 47 55 75 44 51 30 4B 4A 41  9TIG1vZGUuDQ0KJA
41 41 41 41 41 41 41 41 41 59 6D 58 33 67 58 50  AAAAAAAAAYmX3gXP
67 54 73 31 7A 34 45 37 4E 63 2B 42 4F 7A 4A 2B  gTs1z4E7Nc+BOzJ+
51 66 73 31 6A 34 45 37 50 66 35 42 32 7A 54 2F  Qfs1j4E7Pf5B2zT/
67 54 73 37 54 6E 0D 0A 47 62 4E 6D 2B 42 4F 7A  gTs7Tn..GbNm+BOz
50 75 63 41 73 31 58 34 45 37 4E 63 2B 42 4B 7A  PucAs1X4E7Nc+BKz
4A 66 67 54 73 37 54 6E 47 4C 4E 4F 2B 42 4F 7A  JfgTs7TnGLNO+BOz
35 50 34 56 73 31 33 34 45 37 4E 53 61 57 4E 6F  5P4Vs134E7NSaWNo
58 50 67 54 73 77 41 41 41 41 41 41 41 41 41 41  XPgTswAAAAAAAAAA
0D 0A 55 45 55 41 41 45 77 42 42 41 43 34 6A 72  ..UEUAAEwBBAC4jr
63 38 41 41 41 41 41 41 41 41 41 41 44 67 41 41  c8AAAAAAAAAADgAA
38 42 43 77 45 47 41 41 44 41 41 41 41 41 6B 41  8BCwEGAADAAAAAkA
67 41 41 41 41 41 41 46 69 45 41 41 41 41 45 41  gAAAAAAFiEAAAAEA
41 41 41 4E 41 41 41 41 41 41 0D 0A 51 41 41 41  AAANAAAAAA..QAAA
45 41 41 41 41 42 41 41 41 41 51 41 41 41 41 41  EAAAABAAAAQAAAAA
41 41 41 41 42 41 41 41 41 41 41 41 41 41 41 41  AAAABAAAAAAAAAAA
59 41 6B 41 41 42 41 41 41 41 41 41 41 41 41 43  YAkAABAAAAAAAAAC
41 41 41 41 41 41 41 51 41 41 41 51 41 41 41 41  AAAAAAAQAAAQAAAA
41 42 41 41 0D 0A 41 42 41 41 41 41 41 41 41 41  ABAA..ABAAAAAAAA
41 51 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AQAAAAAAAAAAAAAA
41 67 31 67 41 41 5A 41 41 41 41 41 42 51 43 51  Ag1gAAZAAAAABQCQ
41 51 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AQAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 0D 0A  AAAAAAAAAAAAAA..
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 0D 0A 41 4E 41 41 41 4F  AAAAAAAA..ANAAAO
```

```
77 42 41 41 41 41 41 41 41 41 41 41 41 41 41 41  wBAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 75 64 47 56 34 64 41 41 41 41 45  AAAAAudGV4dAAAAE
71 36 41 41 41 41 45 41 41 41 41 4D 41 41 41 41  q6AAAAEAAAAMAAAA
41 51 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41  AQ..AAAAAAAAAAAA
41 41 41 41 41 41 41 67 41 41 42 67 4C 6E 4A 6B  AAAAAAAgAABgLnJk
59 58 52 68 41 41 41 69 45 41 41 41 41 4E 41 41  YXRhAAAiEAAAANAA
41 41 41 67 41 41 41 41 30 41 41 41 41 41 41 41  AAAgAAAA0AAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 0D 0A 51 41  AAAAAAAAAAAA..QA
41 41 51 43 35 6B 59 58 52 68 41 41 41 41 62 46  AAQC5kYXRhAAAAbF
34 49 41 41 44 77 41 41 41 41 55 41 41 41 41 50  4IAADwAAAAUAAAAP
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 45 41 41 41 4D 41 75 63 6E 4E 79 59 77  AAAEAAAMAucnNyYw
41 41 41 42 41 41 0D 0A 41 41 41 41 55 41 6B 41  AAABAA..AAAAUAkA
45 41 41 41 41 41 42 41 41 51 41 41 41 41 41 41  EAAAAABAAQAAAAAA
41 41 41 41 41 41 41 41 41 41 42 41 41 41 42 41  AAAAAAAAAABAAABA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
0D 0A 41 41 41 41 41 41 41 41 41 41 41 41 41 41  ..AAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 0D 0A 41 41 41 41  AAAAAAAAAA..AAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 0D 0A 41 41 41 41 41 41 41 41 41 41  AAAA..AAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 0D 0A  AAAAAAAAAAAAAA..
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 0D 0A 41 41 41 41 41 41  AAAAAAAA..AAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41  AA..AAAAAAAAAAAA
41 41 41 41                                      AAAA

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

[**] Virus - Possible scr Worm [**]
07/13-02:23:57.684488 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x5EA
167.206.112.6:110 -> 46.5.180.250:61658 TCP TTL:58 TOS:0x0 ID:40103 IpLen:20
DgmLen:1500
***A**** Seq: 0x41CC58D9  Ack: 0xAE02C162  Win: 0xFAF0  TcpLen: 20
2B 4F 4B 20 32 37 36 35 20 6F 63 74 65 74 73 0D  +OK 2765 octets.
0A 52 65 74 75 72 6E 2D 70 61 74 68 3A 20 3C 6F  .Return-path: <o
6F 6C 6E 6F 74 69 63 65 40 6F 70 74 6F 6E 6C 69  olnotice@optonli
6E 65 2E 6E 65 74 3E 0D 0A 52 65 63 65 69 76 65  ne.net>..Receive
64 3A 20 66 72 6F 6D 20 6D 74 61 31 31 2E 73 72  d: from mta11.sr
76 2E 68 63 76 6C 6E 79 2E 63 76 2E 6E 65 74 20  v.hcvlny.cv.net
28 6D 74 61 31 31 2E 73 72 76 2E 68 63 76 6C 6E  (mta11.srv.hcvln
79 2E 63 76 2E 6E 65 74 20 5B 31 36 37 2E 32 30  y.cv.net [167.20
36 2E 35 2E 34 36 5D 29 0D 0A 20 62 79 20 6D 73  6.5.46]).. by ms
74 72 32 2E 73 72 76 2E 68 63 76 6C 6E 79 2E 63  tr2.srv.hcvlny.c
76 2E 6E 65 74 0D 0A 20 28 69 50 6C 61 6E 65 74  v.net.. (iPlanet
20 4D 65 73 73 61 67 69 6E 67 20 53 65 72 76 65   Messaging Serve
72 20 35 2E 30 20 50 61 74 63 68 20 32 20 28 62  r 5.0 Patch 2 (b
75 69 6C 74 20 44 65 63 20 31 34 20 32 30 30 30  uilt Dec 14 2000
29 29 0D 0A 20 77 69 74 68 20 45 53 4D 54 50 20  )).. with ESMTP
69 64 20 3C 30 47 4E 55 30 30 47 30 33 48 58 36  id <0GNU00G03HX6
39 30 40 6D 73 74 72 32 2E 73 72 76 2E 68 63 76  90@mstr2.srv.hcv
6C 6E 79 2E 63 76 2E 6E 65 74 3E 20 66 6F 72 0D  lny.cv.net> for.
0A 20 72 6E 70 73 74 65 69 6E 40 69 6D 73 2D 6D  . rnpstein@ims-m
73 2D 64 61 65 6D 6F 6E 20 28 4F 52 43 50 54 20  s-daemon (ORCPT
72 6E 70 73 74 65 69 6E 40 6F 70 74 6F 6E 6C 69  rnpstein@optonli
6E 65 2E 6E 65 74 29 3B 20 54 75 65 2C 0D 0A 20   ne.net); Tue,..
30 34 20 44 65 63 20 32 30 30 31 20 31 39 3A 34   04 Dec 2001 19:4
31 3A 33 30 20 2D 30 35 30 30 20 28 45 53 54 29   1:30 -0500 (EST)
0D 0A 52 65 63 65 69 76 65 64 3A 20 66 72 6F 6D   ..Received: from
20 73 72 76 31 2E 6D 61 69 6C 2E 63 76 2E 6E 65    srv1.mail.cv.ne
74 20 28 73 72 76 31 2E 6D 61 69 6C 2E 63 76 2E   t (srv1.mail.cv.
6E 65 74 20 5B 31 36 37 2E 32 30 36 2E 31 31 32   net [167.206.112
2E 34 30 5D 29 0D 0A 20 62 79 20 6D 74 61 32 2E   .40]).. by mta2.
73 72 76 2E 68 63 76 6C 6E 79 2E 63 76 2E 6E 65   srv.hcvlny.cv.ne
74 0D 0A 20 28 69 50 6C 61 6E 65 74 20 4D 65 73   t.. (iPlanet Mes
73 61 67 69 6E 67 20 53 65 72 76 65 72 20 35 2E   saging Server 5.
30 20 50 61 74 63 68 20 32 20 28 62 75 69 6C 74   0 Patch 2 (built
20 44 65 63 20 31 34 20 32 30 30 30 29 29 0D 0A    Dec 14 2000))..
20 77 69 74 68 20 45 53 4D 54 50 20 69 64 20 3C    with ESMTP id <
30 47 4E 55 30 30 46 46 5A 48 58 35 44 5A 40 6D   0GNU00FFZHX5DZ@m
74 61 32 2E 73 72 76 2E 68 63 76 6C 6E 79 2E 63   ta2.srv.hcvlny.c
76 2E 6E 65 74 3E 20 66 6F 72 0D 0A 20 72 6E 70   v.net> for.. rnp
73 74 65 69 6E 40 6F 70 74 6F 6E 6C 69 6E 65 2E   stein@optonline.
6E 65 74 20 28 4F 52 43 50 54 20 72 6E 70 73 74   net (ORCPT rnpst
65 69 6E 40 6F 70 74 6F 6E 6C 69 6E 65 2E 6E 65   ein@optonline.ne
74 29 3B 20 54 75 65 2C 0D 0A 20 30 34 20 44 65   t); Tue,.. 04 De
63 20 32 30 30 31 20 31 39 3A 34 31 3A 33 32 20   c 2001 19:41:32
2D 30 35 30 30 20 28 45 53 54 29 0D 0A 52 65 63   -0500 (EST)..Rec
65 69 76 65 64 3A 20 66 72 6F 6D 20 70 6F 72 74   eived: from port
61 6C 31 2E 73 72 76 2E 68 63 76 6C 6E 79 2E 63   al1.srv.hcvlny.c
76 2E 6E 65 74 0D 0A 20 28 70 6F 72 74 61 6C 31   v.net.. (portal1
2E 73 72 76 2E 68 63 76 6C 6E 79 2E 63 76 2E 6E   .srv.hcvlny.cv.n
65 74 20 5B 31 36 37 2E 32 30 36 2E 31 31 32 2E   et [167.206.112.
31 33 33 5D 29 0D 0A 20 62 79 20 73 72 76 31 2E   133]).. by srv1.
6D 61 69 6C 2E 63 76 2E 6E 65 74 20 28 53 75 6E   mail.cv.net (Sun
20 49 6E 74 65 72 6E 65 74 20 4D 61 69 6C 20 53    Internet Mail S
65 72 76 65 72 20 73 69 6D 73 2E 34 2E 30 2E 32   erver sims.4.0.2
30 30 30 2E 31 30 2E 31 32 2E 31 36 2E 32 35 2E   000.10.12.16.25.
70 38 29 0D 0A 20 77 69 74 68 20 45 53 4D 54 50   p8).. with ESMTP
20 69 64 20 3C 30 47 4E 55 30 30 45 43 39 48 57    id <0GNU00EC9HW
47 55 55 40 73 72 76 31 2E 6D 61 69 6C 2E 63 76   GUU@srv1.mail.cv
2E 6E 65 74 3E 20 66 6F 72 20 72 6E 70 73 74 65   .net> for rnpste
69 6E 40 6F 70 74 6F 6E 6C 69 6E 65 2E 6E 65 74   in@optonline.net
3B 0D 0A 20 54 75 65 2C 20 30 34 20 44 65 63 20   ;.. Tue, 04 Dec
32 30 30 31 20 31 39 3A 34 31 3A 30 34 20 2D 30   2001 19:41:04 -0
35 30 30 20 28 45 53 54 29 0D 0A 52 65 63 65 69   500 (EST)..Recei
76 65 64 3A 20 28 66 72 6F 6D 20 72 6F 6F 74 40   ved: (from root@
6C 6F 63 61 6C 68 6F 73 74 29 09 62 79 20 70 6F   localhost).by po
72 74 61 6C 31 2E 73 72 76 2E 68 63 76 6C 6E 79   rtal1.srv.hcvlny
2E 63 76 2E 6E 65 74 20 28 38 2E 39 2E 33 2F 38   .cv.net (8.9.3/8
2E 39 2E 33 29 0D 0A 20 69 64 20 54 41 41 32 38   .9.3).. id TAA28
36 37 36 09 66 6F 72 20 72 6E 70 73 74 65 69 6E   676.for rnpstein
40 6F 70 74 6F 6E 6C 69 6E 65 2E 6E 65 74 3B 20   @optonline.net;
54 75 65 2C 20 30 34 20 44 65 63 20 32 30 30 31   Tue, 04 Dec 2001
20 31 39 3A 34 31 3A 32 37 20 2D 30 35 30 30 20    19:41:27 -0500
28 45 53 54 29 0D 0A 44 61 74 65 3A 20 54 75 65   (EST)..Date: Tue
2C 20 30 34 20 44 65 63 20 32 30 30 31 20 31 39   , 04 Dec 2001 19
3A 34 31 3A 32 37 20 2D 30 35 30 30 20 28 45 53   :41:27 -0500 (ES
54 29 0D 0A 46 72 6F 6D 3A 20 6F 6F 6C 6E 6F 74   T)..From: oolnot
69 63 65 40 6F 70 74 6F 6E 6C 69 6E 65 2E 6E 65   ice@optonline.ne
74 0D 0A 53 75 62 6A 65 63 74 3A 20 56 69 72 75   t..Subject: Viru
73 20 41 6C 65 72 74 3A 20 22 67 6F 6E 65 2E 73   s Alert: "gone.s
63 72 22 20 61 74 74 61 63 68 6D 65 6E 74 73 0D   cr" attachments.
0A 54 6F 3A 20 72 6E 70 73 74 65 69 6E 40 6F 70   .To: rnpstein@op
74 6F 6E 6C 69 6E 65 2E 6E 65 74 0D 0A 4D 65 73   tonline.net..Mes
73 61 67 65 2D 69 64 3A 20 3C 32 30 30 31 31 32   sage-id: <200112
30 35 30 30 34 31 2E 54 41 41 32 38 36 37 36 40   050041.TAA28676@
70 6F 72 74 61 6C 31 2E 73 72 76 2E 68 63 76 6C   portal1.srv.hcvl
6E 79 2E 63 76 2E 6E 65 74 3E 0D 0A 4D 49 4D 45   ny.cv.net>..MIME
2D 76 65 72 73 69 6F 6E 3A 20 31 2E 30 0D 0A 43   -version: 1.0..C
6F 6E 74 65 6E 74 2D 74 79 70 65 3A 20 54 45 58   ontent-type: TEX
54 2F 50 4C 41 49 4E 0D 0A 43 6F 6E 74 65 6E 74   T/PLAIN..Content
2D 74 72 61 6E 73 66 65 72 2D 65 6E 63 6F 64 69   -transfer-encodi
6E 67 3A 20 37 42 49 54 0D 0A 0D 0A 41 74 74 65   ng: 7BIT....Atte
6E 74 69 6F 6E 20 4F 70 74 69 6D 75 6D 20 4F 6E   ntion Optimum On
6C 69 6E 65                                       line
```

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

[**] Virus - Possible scr Worm [**]
07/13-02:23:57.694488 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x563
167.206.112.6:110 -> 46.5.180.250:61658 TCP TTL:58 TOS:0x0 ID:40104 IpLen:20
DgmLen:1365
***AP*** Seq: 0x41CC5E8D  Ack: 0xAE02C162  Win: 0xFAF0  TcpLen: 20
20 53 75 62 73 63 72 69 62 65 72 73 3A 0D 0A 56   Subscribers:..V
69 72 75 73 20 41 6C 65 72 74 20 2D 20 57 33 32  irus Alert - W32
2E 47 6F 6E 65 72 2E 41 40 6D 6D 20 41 4B 41 20  .Goner.A@mm AKA
47 6F 6E 65 72 0D 0A 0D 0A 41 20 6E 65 77 20 65  Goner....A new e
6D 61 69 6C 20 77 6F 72 6D 20 6E 61 6D 65 64 20  mail worm named
77 33 32 2E 67 6F 6E 65 72 2E 61 40 6D 6D 20 6F  w32.goner.a@mm o
72 20 22 67 6F 6E 65 72 22 20 68 61 73 20 62 65  r "goner" has be
65 6E 20 69 64 65 6E 74 69 66 69 65 64 20 6F 6E  en identified on
20 74 68 65 20 69 6E 74 65 72 6E 65 74 20 61 6E   the internet an
64 20 63 6F 75 6C 64 20 68 61 76 65 20 61 6E 20  d could have an
69 6D 70 61 63 74 20 6F 6E 20 61 6C 6C 20 69 6E  impact on all in
74 65 72 6E 65 74 20 75 73 65 72 73 2E 0D 0A 0D  ternet users....
0A 54 68 65 20 77 6F 72 6D 20 69 73 20 73 70 72  .The worm is spr
65 61 64 20 61 73 20 61 6E 20 61 74 74 61 63 68  ead as an attach
6D 65 6E 74 20 74 68 72 6F 75 67 68 20 65 6D 61  ment through ema
69 6C 2C 20 74 68 65 20 49 43 51 20 69 6E 73 74  il, the ICQ inst
61 6E 74 20 6D 65 73 73 65 6E 67 65 72 20 70 72  ant messenger pr
6F 67 72 61 6D 2C 20 6F 72 20 49 6E 74 65 72 6E  ogram, or Intern
65 74 20 52 65 6C 61 79 20 43 68 61 74 20 28 49  et Relay Chat (I
52 43 29 2E 0D 0A 0D 0A 54 68 65 20 65 6D 61 69  RC).....The emai
6C 20 70 72 65 73 65 6E 74 73 20 69 74 73 65 6C  l presents itsel
66 20 77 69 74 68 20 74 68 65 20 73 75 62 6A 65  f with the subje
63 74 20 6C 69 6E 65 20 22 48 69 22 20 61 6E 64  ct line "Hi" and
20 74 68 65 20 62 6F 64 79 20 74 65 78 74 20 6F   the body text o
66 2C 20 22 48 6F 77 20 61 72 65 20 79 6F 75 3F  f, "How are you?
20 57 68 65 6E 20 49 20 73 61 77 20 74 68 69 73   When I saw this
20 73 63 72 65 65 6E 20 73 61 76 65 72 2C 20 49   screen saver, I
20 69 6D 6D 65 64 69 61 74 65 6C 79 20 74 68 6F   immediately tho
75 67 68 74 20 61 62 6F 75 74 20 79 6F 75 2E 20  ught about you.
20 49 20 61 6D 20 69 6E 20 61 20 68 61 72 72 79   I am in a harry
2C 20 49 20 70 72 6F 6D 69 73 65 20 79 6F 75 27  , I promise you'
6C 6C 20 6C 6F 76 65 20 69 74 21 22 2E 20 20 54  ll love it!".  T
68 65 20 61 74 74 61 63 68 6D 65 6E 74 20 69 73  he attachment is
20 6E 61 6D 65 64 20 22 67 6F 6E 65 2E 73 63 72   named "gone.scr
22 2E 0D 0A 0D 0A 49 66 20 79 6F 75 20 72 65 63  ".....If you rec
65 69 76 65 20 61 6E 20 65 6D 61 69 6C 20 77 69  eive an email wi
74 68 20 74 68 69 73 20 77 6F 72 64 69 6E 67 20  th this wording
61 6E 64 20 74 68 65 20 22 67 6F 6E 65 2E 73 63  and the "gone.sc
72 22 20 61 74 74 61 63 68 6D 65 6E 74 2C 20 44  r" attachment, D
4F 20 4E 4F 54 20 6F 70 65 6E 20 74 68 65 20 61  O NOT open the a
74 74 61 63 68 6D 65 6E 74 2E 20 20 49 6D 6D 65  ttachment.  Imme
64 69 61 74 65 6C 79 20 64 65 6C 65 74 65 20 74  diately delete t
68 65 20 65 6E 74 69 72 65 20 6D 65 73 73 61 67  he entire messag
65 20 61 6E 64 20 65 6D 70 74 79 20 79 6F 75 72  e and empty your
20 65 6D 61 69 6C 20 74 72 61 73 68 2E 0D 0A 0D   email trash....
0A 41 73 20 61 6C 77 61 79 73 20 77 65 20 61 6C  .As always we al
77 61 79 73 20 72 65 63 6F 6D 6D 65 6E 64 20 74  ways recommend t
68 61 74 20 6F 75 72 20 63 75 73 74 6F 6D 65 72  hat our customer
73 20 72 75 6E 20 61 6E 74 69 76 69 72 75 73 20  s run antivirus
73 6F 66 74 77 61 72 65 2C 20 6B 65 65 70 20 69  software, keep i
74 20 75 70 64 61 74 65 64 20 77 69 74 68 20 74  t updated with t
68 65 20 6C 61 74 65 73 74 20 76 69 72 75 73 20  he latest virus
64 65 66 69 6E 69 74 69 6F 6E 73 2C 20 61 6E 64  definitions, and
20 70 65 72 66 6F 72 6D 20 72 65 67 75 6C 61 72   perform regular
20 73 63 61 6E 73 2E 0D 0A 0D 0A 49 66 20 79 6F   scans.....If yo
75 20 68 61 76 65 20 6F 70 65 6E 65 64 20 74 68  u have opened th
65 20 61 74 74 61 63 68 6D 65 6E 74 20 6F 66 20  e attachment of
61 6E 20 65 6D 61 69 6C 20 74 68 61 74 20 66 69  an email that fi
74 73 20 74 68 69 73 20 64 65 73 63 72 69 70 74  ts this descript
69 6F 6E 2C 20 70 6C 65 61 73 65 20 64 6F 77 6E  ion, please down
6C 6F 61 64 20 66 72 6F 6D 20 79 6F 75 72 20 61  load from your a
6E 74 69 76 69 72 75 73 20 73 6F 66 74 77 61 72  ntivirus softwar
65 20 76 65 6E 64 6F 72 20 74 68 65 20 6C 61 74  e vendor the lat
65 73 74 20 76 69 72 75 73 20 64 65 66 69 6E 69  est virus defini
74 69 6F 6E 73 20 61 6E 64 20 70 65 72 66 6F 72  tions and perfor
```

```
6D 20 61 20 73 63 61 6E 20 6F 66 20 79 6F 75 72  m a scan of your
20 6D 61 63 68 69 6E 65 20 74 6F 20 69 64 65 6E   machine to iden
74 69 66 79 20 74 68 6F 73 65 20 66 69 6C 65 73  tify those files
20 74 68 61 74 20 6E 65 65 64 20 74 6F 20 62 65   that need to be
20 72 65 6D 6F 76 65 64 2E 0D 0A 0D 0A 41 64 64   removed.....Add
69 74 69 6F 6E 61 6C 20 69 6E 66 6F 72 6D 61 74  itional informat
69 6F 6E 20 6F 6E 20 47 6F 6E 65 72 20 6D 61 79  ion on Goner may
20 62 65 20 66 6F 75 6E 64 20 61 74 3A 0D 0A 68   be found at:..h
74 74 70 3A 2F 2F 77 77 77 2E 73 79 6D 61 6E 74  ttp://www.symant
65 63 2E 63 6F 6D 2F 61 76 63 65 6E 74 65 72 2F  ec.com/avcenter/
20 6F 72 20 68 74 74 70 3A 2F 2F 76 69 6C 2E 6D   or http://vil.m
63 61 66 65 65 2E 63 6F 6D 2F 64 65 66 61 75 6C  cafee.com/defaul
74 2E 61 73 70 3F 0D 0A 0D 0A 0D 0A 54 68 61 6E  t.asp?......Than
6B 20 79 6F 75 20 66 6F 72 20 79 6F 75 72 20 61  k you for your a
74 74 65 6E 74 69 6F 6E 20 74 6F 20 74 68 69 73  ttention to this
20 6D 61 74 74 65 72 2E 0D 0A 0D 0A 0D 0A 59 6F   matter.......Yo
75 72 20 4F 70 74 69 6D 75 6D 20 4F 6E 6C 69 6E  ur Optimum Onlin
65 20 54 65 61 6D 0D 0A 0D 0A 2E 0D 0A           e Team.......

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

The first packet was a hit. The content showed that it was part of a mail with attachment XZX900A.scr but a wrong MIME type audio/x-wav. This was the tactics of the worm to exploit a vulnerability in Outlook to cause execution of the code without the user actually clicking on the attachment. We will discuss this later.

The other two packets were parts of a virus alert email that alert a user the spread of Goner worm. Since the email contained the keywords "Goner.scr", IDS alerts were triggered. This turned out to be false positives.

3. Probability the source address was spoofed

No. The mail client was actually making connections to the pop server to retrieve mails.

4. Description of attack

The IDS probably detected a Klez variant that started to spread widely since mid-April.

The worm exploits a vulnerability in Microsoft Outlook and Outlook Express in an attempt to execute itself when you open or even preview the message. Detail information of this vulnerability can be found in Microsoft Security Bulletin (MS01-020).

"Incorrect MIME Header Can Cause IE to Execute E-mail Attachment".
http://www.microsoft.com/technet/security/bulletin/MS01-020.asp

5. Attack mechanism

The virus exploits the following setup:

1/ It embeeds its EXE as a MIME part of the message.
2/ That MIME part is referenced in the IFRAME (that's for

automatically executing through a different application than the mail
client).
3/ The MIME headers of the part containing the message are "forged".
The MIME type is changed to "audio/x-wav" (which should be linked to
the MS Media player) but it has an additional tag indicating that it
wants to be saved as a file with the ".exe" or a ".scr" extention.

The problem is that Outlook uses the MIME type to check what to do
with the inline and it sees that it's a "safe" type (usually
containing only data, no code). But it doesn't know how to read it, so
it calls the associated program and passes it the virus file. Media
player can't recognize the file format (it's a PE, not a WAVE file),
and may try to run the file anyway by sending it to the shell executor
which see that it's an executable and run it.

To further confirm that the attachment was an executable file, I extracted part of the code and make up a text file:

```
Content-Type: multipart/alternative;
boundary=zkVy5s3AIIxh+kVWg)

--zkVy5s3AIIxh+kVWg)
Content-id:<Ih4247XW1TUwi>
Content-type: audio/x-wav;
 name=XZX900A.scr
Content-transfer-encoding: base64

TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAA2AAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5vdCBiZSBydW4gaW4g

--zkVy5s3AIIxh+kVWg)
```

The text file was saved with the name "test.b64". I then used winzip to open "test.b64". Winzip has the ability to open base64 encoded file. I then extracted the file XZX900A.scr from Winzip.

When I open XZX900A.scr with debug, bingo!! This is an executable file
starting with "MZ". (This point thanks to Bill Royds).

```
D:\>debug XZX900A.SCR
-d
15F0:0100  4D 5A 90 00 03 00 00 00-04 00 00 00 FF FF 00 00  MZ..............
15F0:0110  B8 00 00 00 00 00 00 00-40 00 00 00 00 00 00 00  ........@.......
15F0:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
15F0:0130  00 00 00 00 00 00 00 00-00 00 00 00 D8 00 00 00  ................
15F0:0140  0E 1F BA 0E 00 B4 09 CD-21 B8 01 4C CD 21 54 68  ........!..L.!Th
15F0:0150  69 73 20 70 72 6F 67 72-61 6D 20 63 61 6E 6E 6F  is program canno
15F0:0160  74 20 62 65 20 72 75 6E-20 69 6E 20 61 6E 73 66  t be run in ansf
15F0:0170  65 72 2D 65 6E 63 6F 64-69 6E 67 3A 20 62 61 73  er-encoding: bas
-
```

6. Correlations:

Klez is widespread and common in the world. Even though it is now 5
months later since April, Klez is still the top virus according to
anti-virus software vendors and some statistics from press. Here is one
of the references:

Wired - KLEZ: HI MOM, WE'RE NO. 1
http://www.wired.com/news/technology/0,1282,52765,00.html

7. Evidence of active targeting:

Low. Though it is possible that someone delibrately send out viruses to others' mailboxes. In general, the infected victim may not aware of the virus that spread itself through the victim's Internet connection.

8. Severity:

Criticality=3, generally speaking infected machines are personal workstations

Lethality=4, infected machines may send out confidential documents and eventually crash

System countermeasures=?, it depends whether the user has applied up-to-date patches and whether the machine has up-to-date anti-virus software.

Network countermeasures=1,network has not done anything to prevent the spread of worm, except installing an IDS

severity=3+4-?-1=6-?, ranging from 1 - 6, on the high side.

9. Defensive recommendation:

1. Apply up-to-date patches to client machine
2. Install anti-virus software and update signatures frequently
3. Install server side anti-virus system so that virsuses are removed before transmitting to users
4. Modify IDS rule to give a more specific signiture to detect varies worms so as to avoid chance of false positive
5. Install an additional IDS inside the firewall so that the IP address of client machines that received viruses can be reviewed. (This point thanks to Robert Wagner)

10. Multiple choice test question:

What's wrong with the following MIME description:

Content-type: audio/x-wav; name=XZX900A.scr
Content-transfer-encoding: base64.
Content-disposition: attachment filename=XZX900A.scr.

A. audio/x-wav is not a valid MIME type
B. XZX900A.scr is not a valid file name
C. audio/x-wave does not match with base64
D. XZX900A.scr does not match with audio/x-wav

Answer: D
File type ".scr" is an executable in Microsoft Windows, not a media file.

Assignment #3: Analyze This!

Executive Summary

The analysis of this section was based on IDS logs of the University from 1-Aug 2002 to 5-Aug 2002. From the logs, we saw there was a network outage (or IDS outage) at 1-Aug 14:43:30-18:00:02.

At 4-Aug 17:30 an internal host MY.NET.84.234 suddenly generated lot of "Overflow ida" hacking activities. The number of alerts triggered by this host were so great that it accounted for 22% of alerts summing up in 5 days.

An internal host MY.NET.100.208 was attacked by external IP addresses and it was infected by Nimda at 5-Aug 20:50. Tremenous Nimda traffic was then generated by this host which account for almost 65% of alerts of the 5-day period.

There were altogether 53 different alerts detected in this 5-day period. Besides Nimda and IIS worm alerts, there were a lot of Trojan and port scan activities. A few internal hosts were found to be controlled by Trojan and launched external attack.

In a Trojan port scan incident, over 150 internal hosts responded positively. Administrator of the University was advised to check over all those potentially compromised systems.

Logs Analyzed

Alert Logs	Scan Logs	Out-of-spec Logs
alert020801.gz	scans.020801.gz	oos_Aug.1.2002.gz
alert020802.gz	scans.020802.gz	oos_Aug.2.2002.gz
alert020803.gz	scans.020803.gz	oos_Aug.3.2002.gz
alert020804.gz	scans.020804.gz	oos_Aug.4.2002.gz
alert020805.gz	scans.020805.gz	oos_Aug.5.2002.gz

After fixing some minor problems of the alert files, they were catencated to a large file, and those "spp_portscan" entries were removed. There were total 2,243,643 alerts in these five days. The following is a table showing the distribution of these alerts in each hour.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	840	0.04%	1268	0.06%	596	0.03%	2062	0.09%	1161	0.05%
01	776	0.03%	1155	0.05%	707	0.03%	1675	0.07%	1282	0.06%
02	831	0.04%	1195	0.05%	496	0.02%	1043	0.05%	1439	0.06%
03	693	0.03%	1354	0.06%	669	0.03%	1193	0.05%	970	0.04%
04	641	0.03%	2084	0.09%	2909	0.13%	1756	0.08%	901	0.04%
05	734	0.03%	1474	0.07%	414	0.02%	1252	0.06%	998	0.04%
06	1800	0.08%	1418	0.06%	558	0.02%	1212	0.05%	1776	0.08%
07	7869	0.35%	4354	0.19%	1522	0.07%	1683	0.08%	1394	0.06%
08	8370	0.37%	3011	0.13%	29150	1.30%	1761	0.08%	2655	0.12%
09	13269	0.59%	2626	0.12%	2964	0.13%	2658	0.12%	2913	0.13%
10	9357	0.42%	2983	0.13%	1259	0.06%	3797	0.17%	10954	0.49%
11	6479	0.29%	1347	0.06%	1420	0.06%	2860	0.13%	4214	0.19%
12	4854	0.22%	1290	0.06%	1406	0.06%	5482	0.24%	2095	0.09%
13	0	0.00%	3327	0.15%	1368	0.06%	2621	0.12%	8931	0.40%
14	0	0.00%	3279	0.15%	3271	0.15%	2298	0.10%	5145	0.23%
15	0	0.00%	2583	0.12%	1086	0.05%	1622	0.07%	6169	0.27%
16	0	0.00%	1784	0.08%	2333	0.10%	1343	0.06%	19644	0.88%
17	0	0.00%	1009	0.04%	10687	0.48%	114021	5.08%	1041	0.05%
18	2027	0.09%	517	0.02%	2622	0.12%	244595	10.90%	2248	0.10%
19	1749	0.08%	1193	0.05%	1572	0.07%	129378	5.77%	1486	0.07%
20	1662	0.07%	1752	0.08%	748	0.03%	1832	0.08%	3937	0.18%
21	1207	0.05%	1211	0.05%	1352	0.06%	1764	0.08%	334362	14.90%
22	1455	0.06%	1393	0.06%	1319	0.06%	1294	0.06%	585347	26.09%
23	1694	0.08%	910	0.04%	2202	0.10%	1586	0.07%	528339	23.55%

Table 1. Distribution of alerts over the 5-day period.

I will use this kind of distribution table through out the whole analysis of this paper. Incidents of interest were highlighted in red color. Highlights were based on extra ordinary number of alerts happened in a hour when compared to other hours of the day.

Let's look at some incidents in the above table. There were no IDS logs at hours 13-17 at 1-Aug. A closer look at the log files indicate that outage of IDS logs was between 14:43:30-18:00:02. We do not know exactly what happen at that time, but there were evidence that the outage of IDS logs may be due to some network issues.

Another incident occurred at hours 17-19 at 4-Aug. An internal host MY.NET.84.234 suddenly got crazy at 17:30 and generating a lot of "Overflow ida" hacking activities.

The third incident was the most serious one. An internal host MY.NET.100.208 attacked by three IP addresses at the same time and one successfully infected it Nimda at 20:50. Tremenous Nimda traffic was then generated by this IP. There were total 1,444,138 alerts related to this single IP addresswhich account for almost 65% of alerts of five days.

We will discuss the above three incidents in greater details later. There were other minor incidents like port scan, Trojan control, etc. All these will be discussed later.

There were total 53 different alerts identified in the 5-day period. They are sorted according to number of occurence below:

880709	39.25%	NIMDA - Attempt to execute cmd from campus host
495809	22.10%	spp_http_decode: IIS Unicode attack detected
483710	21.56%	IDS552/web-iis_IIS ISAPI Overflow ida INTERNAL nosize
123862	5.52%	NIMDA - Attempt to execute root from campus host
106946	4.77%	UDP SRC and DST outside network
53565	2.39%	spp_http_decode: CGI Null Byte attack detected
30099	1.34%	SMB Name Wildcard
24223	1.08%	TFTP - External UDP connection to internal tftp server
14579	0.65%	External RPC call
11923	0.53%	Watchlist 000220 IL-ISDNNET-990517
4114	0.18%	Possible trojan server activity
2543	0.11%	SUNRPC highport access!
2055	0.09%	IRC evil - running XDCC
1306	0.06%	Watchlist 000222 NET-NCFC
1293	0.06%	EXPLOIT x86 NOOP
1120	0.05%	Queso fingerprint
927	0.04%	SNMP public access
788	0.04%	connect to 515 from outside
731	0.03%	Attempted Sun RPC high port access
679	0.03%	Samba client access
628	0.03%	High port 65535 udp - possible Red Worm - traffic
316	0.01%	IDS552/web-iis_IIS ISAPI Overflow ida nosize
260	0.01%	ICMP SRC and DST outside network
236	0.01%	SMB C access
173	0.01%	TFTP - Internal UDP connection to external tftp server
166	0.01%	beetle.ucs
147	0.01%	Port 55850 tcp - Possible myserver activity - ref. 010313-1
136	0.01%	Incomplete Packet Fragments Discarded
106	0.00%	Null scan!
88	0.00%	NMAP TCP ping!
58	0.00%	EXPLOIT x86 setuid 0
53	0.00%	Tiny Fragments - Possible Hostile Activity
48	0.00%	EXPLOIT x86 stealth noop
44	0.00%	High port 65535 tcp - possible Red Worm - traffic
42	0.00%	STATDX UDP attack
38	0.00%	EXPLOIT x86 setgid 0

18	0.00%	Port 55850 udp - Possible myserver activity - ref. 010313-1
13	0.00%	SMB CD...
13	0.00%	TCP SRC and DST outside network
11	0.00%	MY.NET.30.4 activity
11	0.00%	External FTP to HelpDesk MY.NET.70.50
11	0.00%	HelpDesk MY.NET.70.50 to External FTP
9	0.00%	HelpDesk MY.NET.70.49 to External FTP
8	0.00%	External FTP to HelpDesk MY.NET.70.49
6	0.00%	TFTP - External TCP connection to internal tftp server
5	0.00%	EXPLOIT NTPDX buffer overflow
4	0.00%	HelpDesk MY.NET.83.197 to External FTP
3	0.00%	Back Orifice
3	0.00%	DDOS shaft client to handler
3	0.00%	RFB - Possible WinVNC - 010708-1
2	0.00%	SYN-FIN scan!
2	0.00%	Traffic from port 53 to port 123
1	0.00%	MY.NET.30.3 activity

Table 2. Alerts detected in 5-day period and sorted in decending order

We will start discussing the top 10 alerts and a few specially selected alerts.

Nimda Attack

Nimda was found on September 18th, 2001. It quickly spread around the world. It is a complex virus with a mass mailing worm component which spreads itself in attachments named README.EXE. If affects Windows 95, Windows 98, Windows Me, Windows NT 4 and Windows 2000 users. Analyses indicated that the worm attempts to propagate itself to new victims via four distinct mechanisms.

1. The worm scans the Internet looking for IIS servers and attempts to exploit a number of IIS vulnerabilities to gain control of a victim host. Network attacks include exploitation of the "IIS Directory Traversal Vulnerability", and utilization of backdoors left behind by previous Code Red II and Sadmind infections. Once in control of a victim IIS server, the worm uses TFTP to transfer its code from the attacking machine to the victim. The file transferred via TFTP is named Admin.dll.

2. The worm harvests email addresses from the Windows address book and user's inboxes and sends itself to all addresses as an attachment named "readme.exe". Note that any x86 email software that uses Internet Explorer 5.5 SP1 or earlier to display HTML messages will automatically execute the malicious attachment if the message is merely opened or previewed. This happens because the worm MIME encodes the attachment to take advantage of a known vulnerability called "Automatic Execution of Embedded MIME Types" (see CERT advisory CA-2001-06). Microsoft's Outlook and Outlook Express are the most typical victims.

3. If the worm successfully infects a web server, it uses the HTTP service to propagate itself to clients who browse the web server's pages. Upon infecting a victim server, the worm creates a copy of itself named "readme.eml" and traverses the directory tree (including network shares) searching for web-related files such as those with .html, .htm, or .asp extensions. Each time the worm finds a web content file, it appends a piece of JavaScript to the file. The JavaScript forces a download of readme.eml to any client that views the file via a browser. Some versions of Internet Explorer will automatically execute the readme.eml file and allow the worm to infect the client. The IE vulnerability issue here is the same as in the email propagation mechanism; that is, IE 5.5 SP1 or earlier is vulnerable to the "Automatic Execution of Embedded MIME Types" problem. Allowing JavaScript in the browser enables the attack to take advantage of the vulnerability.

4. The worm is network aware and propagates via open file shares. It will copy itself to all directories, including those found on a network share, for which the user has write permission. These worm copies are named "readme.eml". Any other host that accesses the share and executes or previews one of these files can become infected.

The alerts of the IDS here mainly were generated by the first kind of Nimda infecting activities, where the infected machine try to attack port 80 of other potential vulnerable web servers. And tftp was used to transfer the virus body from the attacker machine to the victim machine.

The two Nimda related alerts "NIMDA - Attempt to execute cmd from campus host" and "NIMDA - Attempt to execute root from campus host" altogether account for almost 45% of total alerts collected in the 5-day period. I do not find these two alerts in the previous work of other GCIA analysts. I guess the University may have newly created these rules by modifying a more generic one. I noted that no Nimda alerts were found initiated from other Internal hosts. The two alerts may be generated by snort rules like these:

```
alert tcp $INTERNAL_NET any -> any 80 (msg: "NIMDA - Attempt to execute
cmd from campus host"; flags: A+; content: "cmd.exe"; nocase;)

alert tcp $INTERNAL_NET any -> any 80 (msg: "NIMDA - Attempt to execute
root from campus host"; flags: A+; content: "root.exe"; nocase;)
```

By looking at the distribution below, it was noticed that most of the NIMDA alerts was due to infection of internal host MY.NET.100.208 at Aug-5 night.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
01	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
02	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
05	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
09	0	0.00%	0	0.00%	0	0.00%	0	0.00%	1	0.00%
10	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
11	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
12	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
13	0	0.00%	1	0.00%	0	0.00%	0	0.00%	1	0.00%
14	0	0.00%	0	0.00%	0	0.00%	2	0.00%	1	0.00%
15	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
16	0	0.00%	0	0.00%	0	0.00%	0	0.00%	1	0.00%
17	0	0.00%	1	0.00%	0	0.00%	0	0.00%	0	0.00%
18	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
19	0	0.00%	0	0.00%	1	0.00%	0	0.00%	0	0.00%
20	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
21	0	0.00%	0	0.00%	0	0.00%	0	0.00%	231302	10.31%
22	0	0.00%	0	0.00%	0	0.00%	0	0.00%	406115	18.10%
23	0	0.00%	0	0.00%	0	0.00%	0	0.00%	367173	16.37%

Table 3. Distribution of Nimda alerts

All Nimda traffic are generated by internal hosts.

Occurence	Internal Host	Remarks
1004562	MY.NET.100.208	Nimda infected from 211.141.120.18
1	MY.NET.105.10	Nimda attempt to 65.54.250.120, many IIS stuff
1	MY.NET.111.30	Nimda attempt to 207.46.235.150, many IIS stuff
1	MY.NET.130.20	Nimda attempt to 65.54.250.121, many SNMP public access from 131.118.250.157,131.118.250.143 & 131.118.250.129
1	MY.NET.165.19	Nimda attempt to 65.54.250.120
1	MY.NET.70.144	Nimda attempt to 207.46.235.150
1	MY.NET.70.16	Nimda attempt to 65.54.250.121, many other IIS stuff
1	MY.NET.70.169	Nimda attempt to 65.54.250.121, many other IIS stuff
1	MY.NET.82.87	IRC initiated to 66.250.104.241, may SMB traffic
1	MY.NET.83.176	remote by Trojan at 217.136.63.141

Table 4. Internal hosts that generated Nimda traffic

Analysis of Nimda infection of MY.NET.100.208

MY.NET.85.100.208 began to show abnormal tftp outbound traffic at Aug-5 15:47. TFTP has been used by Nimda to transfer the virus body from the attacker to the newly infected machine. In the log here, I don't see the how this IP got infected, may be due to the fact that the IDS did not logged Nimda traffic from external hosts. However, outbound TFTP target (192.168.0.1) was bogus. The reason for this bogus IP was not clear. May be these few attacks were coming from a host that had internal IP 192.168.0.1 which was translated to a valid external IP when going out to the Internet. The results of the attack to MY.NET.100.208 caused the host trying to connect back to 192.168.0.1.

```
08/05-15:09:28.532727  [**] connect to 515 from outside [**] 216.241.23.211:3366 -> MY.NET.100.208:515
08/05-15:47:14.743794  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1105 -> 192.168.0.1:69
08/05-15:47:15.737438  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1105 -> 192.168.0.1:69
```

At 15:49, the host was hit by another IP 61.189.144.7 and an "IIS unicode attack" was detected. The host then show another stream of bogus tftp outbound attempts to 169.254.126.97 which is in the range of Microsoft's APIPA (Automatic Private IP Addressing). APIPA uses the range 169.254.0.1 through 169.254.255.254 with a subnet mask 255.255.0.0.

```
08/05-15:49:03.249785  [**] spp_http_decode: IIS Unicode attack detected [**] 61.189.144.7:4935 -> MY.NET.100.208:80
08/05-15:49:03.249785  [**] spp_http_decode: IIS Unicode attack detected [**] 61.189.144.7:4935 -> MY.NET.100.208:80
08/05-15:49:03.249785  [**] spp_http_decode: IIS Unicode attack detected [**] 61.189.144.7:4935 -> MY.NET.100.208:80
……..
08/05-17:41:06.786147  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1296 ->
169.254.126.97:69
08/05-17:41:07.806161  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1299 ->
169.254.126.97:69
```

At 15:50, another IP 61.145.69.74 hit the host with "IIS Unicode attack" again. This time we see a tftp outbound connection back to the host 61.145.69.74.

```
08/05-20:49:06.258802  [**] spp_http_decode: IIS Unicode attack detected [**] 61.145.69.74:3912 -> MY.NET.100.208:80
08/05-20:49:06.936641  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1915 -> 61.145.69.74:69
08/05-20:44:29.515759  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1910 -> 61.145.69.74:69
08/05-20:44:29.517714  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1910 -> 61.145.69.74:69
08/05-20:44:30.208084  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1911 -> 61.145.69.74:69
```

However, the infection seemed still not successful. Until 21:21, another IP address 211.141.120.18 hit the host again. The host then was successfully infected by Nimda and start to generate tremenous Nimda traffic.

```
0/05-21:21:06.650638  [**] spp_http_decode: IIS Unicode attack detected [**] 211.141.120.18:4853 -> MY.NET.100.208:80
08/05-21:21:06.650638  [**] spp_http_decode: IIS Unicode attack detected [**] 211.141.120.18:4853 -> MY.NET.100.208:80
…..
08/05-21:21:11.439315  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1930 ->
211.141.120.18:69
08/05-21:21:11.590781  [**] TFTP - Internal UDP connection to external tftp server [**] MY.NET.100.208:1930 ->
211.141.120.18:69
….
08/05-21:21:55.661920  [**] NIMDA - Attempt to execute root from campus host [**] MY.NET.100.208:2008 -> 130.95.40.191:80
08/05-21:21:55.664339  [**] NIMDA - Attempt to execute root from campus host [**] MY.NET.100.208:2010 -> 130.7.64.55:80
```

By 23:59, there were about 1,004,590 Nimda alerts generated by MY.NET.100.208 and about 110,189 unique IP addresses were attacked. Immediately action needed to be taken to stop this machine.

Referring to table 4. There were 9 other hosts that triggered Nimda alert. However, to another extreme, each host only triggered the alert once. As Nimda infected machines always generate Nimda attack traffic continuously, these alerts may be false-alarms. 5 out of the 9 IP addresses (MY.NET.105.10,MY.NET.130.20,MY.NET.165.19,MY.NET.70.16,MY.NET.70.169) had Nimda alert targeted against 65.54.250.12x. A whois search on 65.54.250.12x indicated that these are IP addresses belonged to Microsoft. It was highly possible that web browing of some users triggered this alert. 2 out of the 9 IP addresses (MY.NET.70.16,MY.NET.70.169) had Nimda alert targetted against 207.46.235.150. This IP address also belonged to Microsoft.

For MY.NET.82.87, there were Nimda alert targetted against 207.46.235.162 (Microsoft's) and IRC alerts targetted 66.250.104.241:6667.

```
08/05-08:04:21.131683  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-08:12:20.217293  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-08:22:20.222372  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-08:42:20.197729  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-08:52:20.197432  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-09:02:20.195997  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
08/05-09:32:20.520276  [**] IRC evil - running XDCC [**] MY.NET.82.87:1053 -> 66.250.104.241:6667
```

A reverse lookup of the IP address 66.250.104.241 yielded alpha.aggressivehosting.net. It was unusual that a hosting company hosted a IRC server.
A search on www.dshield.org for this IP address (66.250.104.241) indicated that there had been 76 records against this IP address and involved 26 targets. Ports attacked by this IP address included 1080, 3128, 8080 and 80.

It was very likely that the host MY.NET.82.87 was compromised. Administrator of the University should approach the user of the host and have a check on that.

Another IP MY.NET.83.176 also made connection to Microsoft's IP which triggered Nimda alert. However, after searching the 5-day alert, this host seemed to be Trojaned by Sub-Seven and remotely controlled by IP 217.136.63.141. A search on www.dshield.org yielded no result on this IP address.

```
08/03-19:55:53.607645  [**] NIMDA - Attempt to execute cmd from campus host [**] MY.NET.83.176:1345 -> 207.68.132.9:80
08/04-14:00:41.976096  [**] Possible trojan server activity [**] 217.136.63.141:4312 -> MY.NET.83.176:27374
08/04-14:00:41.976378  [**] Possible trojan server activity [**] MY.NET.83.176:27374 -> 217.136.63.141:4312
08/04-14:00:42.556520  [**] Possible trojan server activity [**] 217.136.63.141:4312 -> MY.NET.83.176:27374
08/04-14:00:42.556798  [**] Possible trojan server activity [**] MY.NET.83.176:27374 -> 217.136.63.141:4312
08/04-14:00:43.161432  [**] Possible trojan server activity [**] 217.136.63.141:4312 -> MY.NET.83.176:27374
08/04-14:00:43.161659  [**] Possible trojan server activity [**] MY.NET.83.176:27374 -> 217.136.63.141:4312
```

Recommendations: Administrator of the University should follow up with the hosts listed in Table 4.

IIS Unicode Attack

This alert was always one of the top three alerts in all the work of other GCIA analysts. We have altogether 495,809 alerts here that accounted for 22% of alerts of the 5-day period. This alert was generated by snort http_decode preprocessor which look out for Unicode-encoded "\" "/" and "." characters.

The Unicode attack is not new, but rather a variation on an old vulnerability called the “Dot Dot” attack. The Dot Dot attack occurs when an attacker sends a malformed URL to a web server that contained "../../". The “../” tells the web server to look up one directory. The number of “../” ‘s does not matter as long as there are enough of them to recurse back to the root of the file system (either c:\ or / on Unix systems). The IIS Unicode attack uses the http protocol and malformed URLs to traverse directories and execute arbitrary commands on vulnerable web servers, much like the “Dot Dot” attack. The IIS Unicode attack uses a Unicode representation of a directory delimiter (/) to fool IIS into doing the same thing as the old Dot Dot attack. The fix to the Dot Dot attack does not recognize the Unicode representation of the slash, which is why this attack works.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	195	0.01%	431	0.02%	41	0.00%	451	0.02%	28	0.00%
01	45	0.00%	505	0.02%	41	0.00%	69	0.00%	16	0.00%
02	25	0.00%	392	0.02%	10	0.00%	53	0.00%	94	0.00%
03	30	0.00%	496	0.02%	29	0.00%	27	0.00%	25	0.00%
04	32	0.00%	680	0.03%	23	0.00%	45	0.00%	27	0.00%
05	35	0.00%	496	0.02%	13	0.00%	348	0.02%	46	0.00%
06	43	0.00%	516	0.02%	18	0.00%	55	0.00%	47	0.00%
07	249	0.01%	467	0.02%	33	0.00%	64	0.00%	160	0.01%
08	126	0.01%	150	0.01%	66	0.00%	51	0.00%	1400	0.06%
09	812	0.04%	818	0.04%	93	0.00%	77	0.00%	1990	0.09%
10	2186	0.10%	2367	0.11%	71	0.00%	36	0.00%	6586	0.29%
11	1171	0.05%	457	0.02%	395	0.02%	91	0.00%	3448	0.15%
12	540	0.02%	471	0.02%	634	0.03%	459	0.02%	1126	0.05%
13	0	0.00%	1403	0.06%	550	0.02%	915	0.04%	776	0.03%
14	0	0.00%	750	0.03%	950	0.04%	250	0.01%	1038	0.05%
15	0	0.00%	1020	0.05%	217	0.01%	452	0.02%	1047	0.05%
16	0	0.00%	887	0.04%	201	0.01%	305	0.01%	976	0.04%
17	0	0.00%	407	0.02%	215	0.01%	812	0.04%	115	0.01%
18	598	0.03%	63	0.00%	109	0.00%	437	0.02%	160	0.01%
19	630	0.03%	260	0.01%	89	0.00%	397	0.02%	476	0.02%
20	716	0.03%	426	0.02%	44	0.00%	500	0.02%	2984	0.13%
21	550	0.02%	39	0.00%	36	0.00%	178	0.01%	101777	4.54%
22	513	0.02%	323	0.01%	37	0.00%	98	0.00%	177975	7.93%
23	623	0.03%	11	0.00%	253	0.01%	69	0.00%	160157	7.14%

Table 5. Distribution of IIS Unicode Attack alerts in 5-day period.

To certain extent, this alert was quite noisy. There were altogether 572 distinct hosts that triggered this alert and among them 229 were hosts of the University.

The top 20 hosts that triggered this alert:

Occurrence	%	IP Address	Remarks
439389	19.58 %	MY.NET.100.208	Nimda infected. Already discussed
6986	0.31%	MY.NET.85.74	Target against 207.200.86.97 from 1-Aug to 5-Aug, target against 64.12.147.25 at 5-Aug as well, showed signs of Trojan
6895	0.31%	80.137.90.34	Account for the incident at 5-Aug 10:00. Target against MY.NET.5.14. Stopped after generating 75min traffic.
2885	0.13%	MY.NET.152.19	Target against many hosts at 211.x.x.x, showed signs of Trojan
2828	0.13%	MY.NET.153.145	Target against many hosts at 211.x.x.x.
2475	0.11%	151.203.178.36	Attack to many internal hosts at 5-Aug 20:38
2003	0.09%	MY.NET.153.168	Show web browsing activities to 212.179.x.x (ISDNNET), attack to 211.x.x.x, showed signs of Trojan
1855	0.08%	MY.NET.153.143	Target against many hosts at 211.x.x.x
1642	0.07%	MY.NET.91.103	Target against many hosts at 211.x.x.x
1592	0.07%	MY.NET.91.105	Target against many hosts at 211.x.x.x
1143	0.05%	MY.NET.15.212	Target against 64.12.x.x and some other hosts
1108	0.05%	MY.NET.88.143	Target against many hosts at 211.x.x.x
966	0.04%	MY.NET.111.196	Target against 64.12.x.x and others
930	0.04%	MY.NET.153.195	Target against many hosts at 211.x.x.x, showed signs of Trojan
655	0.03%	MY.NET.153.146	Target against many 211.x.x.x hosts
634	0.03%	MY.NET.178.57	Target against 64.12.x.x and 207.200.x.x
548	0.02%	MY.NET.190.16	Target against 64.12.x.x and others
544	0.02%	MY.NET.153.121	Target against many 211.x.x.x hosts and others
506	0.02%	MY.NET.153.194	Target against many 211.x.x.x hosts and others, showed signs of Trojan
503	0.02%	202.98.223.86	Target against many internal hosts at 3-Aug 11:00 - 13:00

Table 6. Top 20 hosts that triggered IIS Unicode Attack alerts

There were other 212 internal hosts generating this IIS Unicode alerts besides the 17 internal hosts listed in the table. It was quite difficult to believe that over 200 internal hosts were compromised and generating these IIS Unicode attack at the same period of time. By looking at the distribution table, one will notice that this alert was in fact quite noisy. Many of these alerts may be false postives. However, some of the internal hosts really showed signs of Trojan. One important job of the analyst is to identify those hosts that were really compromised and generating these alerts. Asking the administrator to check over all the 229 internal hosts that generated these alerts may be unrealistic.

After removing IIS Unicode alerts from My.NET.100.208, 151.203.178.36, 80.137.90.34 and 202.98.223.86, we got the below top 20 mostly IIS Unicode targetted hosts.

Alerts	Target IP Address	DNS Host Name	Remarks
4758	207.200.86.97	myns-v1.websys.aol.com	English content
3249	207.200.86.66	myns-v2.websys.aol.com	English content
1364	211.233.27.52	none	Korea content
1087	207.200.89.193	home-v1.websys.aol.com	English content
936	211.169.241.204	none	Cannot load index.html but IP in Korea Netblock
720	211.233.27.74	none	Korea content
693	64.12.51.118	img-mv1.websys.aol.com	English content
693	211.115.213.202	none	Cannot load index.html but IP in Korea Netblock
672	64.12.42.116	cdn-v03.websys.aol.com	English content
612	211.239.123.75	none	Korea content
587	64.12.48.217	eventfarm3-vip.ptn.aol.com	English content
558	203.248.242.131	ftp.webhard.co.kr	Korea content
526	205.188.237.183	cdn-v05.stream.aol.com	English content
515	64.12.42.117	cdn-v04.websys.aol.com	English content
478	211.115.213.207	none	Cannot load index.html but IP in Korea Netblock
468	199.244.218.42	www.capitalone.com	English content
436	211.233.29.224	none	Korea content
434	211.32.117.38	www18.hanmail.net	Korea content
397	211.233.28.118	none	Korea content
381	211.32.117.39	www19.hanmail.net	Korea content

Table 7. Top 20 IIS Unicode Attack alerts target hosts

It was quite impressive that half of the top target IP addresses were Korea's IP addresses. Since Korea's language character sets are implemented with unicode (other examples are China and Taiwan), it was highly possible that false postive alerts were triggered by users browsing to these web sites. Other xxx.aol.com web sites actually provide web mail & personalized portal services. These sites use a lot of cookies, form parameters, etc. which easily triggered Unicode alerts.

Recommendations: It was suggested that the University either disable unicode alerts or remove those alerts in log files targetted again Korea IP addresses and aol.com to avoid such a high false positive rate during intrusion analysis.

IDS552/web-iis_IIS ISAPI Overflow ida INTERNAL nosize

This alerts accounted for over 20% of total alerts observed in the 5-day interval. The alert indicated that a remote attacker has attempted to exploit a vulnerability in Microsoft IIS. An unchecked buffer in the Microsoft IIS Index Server ISAPI Extension could enable a remote intruder to gain SYSTEM access to the web server.

The snort rule was something like this:

```
alert TCP $EXTERNAL any -> $INTERNAL 80 (msg: "IDS552/web-iis_IIS ISAPI
Overflow ida"; dsize: >239; flags: A+; content: ".ida?";)
```

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
01	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
02	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
05	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
09	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
10	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
11	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
12	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
13	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
14	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
15	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
16	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
17	0	0.00%	0	0.00%	0	0.00%	112298	5.01%	0	0.00%
18	0	0.00%	0	0.00%	0	0.00%	243191	10.84%	0	0.00%
19	0	0.00%	0	0.00%	0	0.00%	128221	5.71%	0	0.00%
20	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
21	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
22	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
23	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

Table 8. Distribution of ISAPI Overflow IDA alerts in 5-day period

From the above distribution table, we clearly saw that there was an attack occurred at 4-Aug 17:xx-19:xx. By looking at the log, the exact time for the event was 17:30:00-19:29:34. All 483710 alerts were generated by a single internal host MY.NET.84.234 and 481853 victims were attacked.

This host was probably compromised and there were signs of Trojan and was controlled by 80.62.155.240. This hacker IP also controlled some other internal hosts as well and will be discussed again in the section of Trojan alert.

```
08/02-09:05:25.229069  [**] Possible trojan server activity [**] 80.62.155.240:1715 -> MY.NET.84.234:27374
08/02-09:05:25.229329  [**] Possible trojan server activity [**] MY.NET.84.234:27374 -> 80.62.155.240:1715
08/02-09:05:25.822492  [**] Possible trojan server activity [**] 80.62.155.240:1715 -> MY.NET.84.234:27374
08/02-09:05:25.822735  [**] Possible trojan server activity [**] MY.NET.84.234:27374 -> 80.62.155.240:1715
```

UDP SRC and DST outside network

This alert account for 4.77% of the total alerts in the 5-days interval. Here is the distribution table.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	256	0.01%	399	0.02%	283	0.01%	611	0.03%	609	0.03%
01	318	0.01%	324	0.01%	317	0.01%	576	0.03%	639	0.03%
02	341	0.02%	387	0.02%	251	0.01%	459	0.02%	665	0.03%
03	283	0.01%	459	0.02%	300	0.01%	453	0.02%	464	0.02%
04	272	0.01%	651	0.03%	210	0.01%	437	0.02%	413	0.02%
05	339	0.02%	479	0.02%	200	0.01%	483	0.02%	469	0.02%
06	442	0.02%	490	0.02%	240	0.01%	580	0.03%	671	0.03%
07	6123	0.27%	3465	0.15%	537	0.02%	615	0.03%	606	0.03%
08	7680	0.34%	2214	0.10%	724	0.03%	643	0.03%	553	0.02%
09	6467	0.29%	665	0.03%	651	0.03%	730	0.03%	429	0.02%
10	5598	0.25%	255	0.01%	604	0.03%	731	0.03%	359	0.02%
11	4431	0.20%	522	0.02%	519	0.02%	685	0.03%	298	0.01%
12	3744	0.17%	298	0.01%	395	0.02%	696	0.03%	310	0.01%
13	0	0.00%	298	0.01%	412	0.02%	719	0.03%	249	0.01%
14	0	0.00%	294	0.01%	448	0.02%	648	0.03%	264	0.01%
15	0	0.00%	270	0.01%	417	0.02%	632	0.03%	291	0.01%
16	0	0.00%	328	0.01%	449	0.02%	550	0.02%	18241	0.81%
17	0	0.00%	212	0.01%	611	0.03%	352	0.02%	408	0.02%
18	254	0.01%	212	0.01%	731	0.03%	537	0.02%	345	0.02%
19	206	0.01%	269	0.01%	724	0.03%	379	0.02%	364	0.02%
20	404	0.02%	488	0.02%	371	0.02%	648	0.03%	386	0.02%
21	315	0.01%	521	0.02%	734	0.03%	682	0.03%	510	0.02%
22	454	0.02%	600	0.03%	649	0.03%	581	0.03%	461	0.02%
23	530	0.02%	331	0.01%	704	0.03%	678	0.03%	468	0.02%

Table 9. Distribution of UDP SRC and DST outside network alerts in 5-day period

This alert seemed to be tailor-made rule added by the University's administrator as I cannot find similar rule in standard snort package. By looking at the distribution table, the alerts were quite evenly distributed through out the days except two or three special incidents. May be these alerts were generated by faulty network equipment?

Everyday , there were 0.01%-0.02% alerts per hour due to udp packets from 3.0.0.99:137 to 10.0.0.1:137.

```
08/01-00:00:05.732948  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:08.733589  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:11.738004  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:13.240134  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:14.742414  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:16.244524  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:19.248938  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:23.755940  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:25.257650  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
08/01-00:00:26.759844  [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
```

The red incident highlighted in distribution table in 1-Aug, was due to a tremoneous stream of udp packets from 63.250.213.12:1031 to 233.28.65.148:5779. The red incident highlighted in 2-Aug was due to another stream of udp packets from 63.250.213.73:1033 -> 233.28.65.173:5779. However, in the red incident at 5-Aug, the source IP addresses were quite unpredictable but the destination IP address were fixed.

```
08/05-16:40:33.432956  [**] UDP SRC and DST outside network [**] 131.94.133.48:1025 -> 233.2.171.1:56464
08/05-16:40:33.436597  [**] UDP SRC and DST outside network [**] 140.221.8.53:32826 -> 233.2.171.1:56464
08/05-16:40:33.438017  [**] UDP SRC and DST outside network [**] 129.215.0.201:32769 -> 233.2.171.1:56464
08/05-16:40:33.445208  [**] UDP SRC and DST outside network [**] 192.12.209.24:32771 -> 233.2.171.1:56464
08/05-16:40:33.446396  [**] UDP SRC and DST outside network [**] 198.49.215.220:1025 -> 233.2.171.1:56464
08/05-16:40:33.447527  [**] UDP SRC and DST outside network [**] 131.193.77.111:2338 -> 233.2.171.1:56464
08/05-16:40:33.447680  [**] UDP SRC and DST outside network [**] 198.49.215.220:1027 -> 233.2.171.1:56464
08/05-16:40:33.450551  [**] UDP SRC and DST outside network [**] 129.24.32.209:1039 -> 233.2.171.1:56464
08/05-16:40:33.451295  [**] UDP SRC and DST outside network [**] 128.2.184.130:4247 -> 233.2.171.1:56464
08/05-16:40:33.452116  [**] UDP SRC and DST outside network [**] 199.104.137.2:32908 -> 233.2.171.1:56464
08/05-16:40:33.454276  [**] UDP SRC and DST outside network [**] 129.69.9.126:32881 -> 233.2.171.1:56464
08/05-16:40:33.455137  [**] UDP SRC and DST outside network [**] 129.237.25.150:1029 -> 233.2.171.1:56464
08/05-16:40:33.455846  [**] UDP SRC and DST outside network [**] 128.2.6.32:32792 -> 233.2.171.1:56464
```

Since the content of these UDP packets were not available in the log files, it was quite difficult to determine what these UDP packets actually were. But there was a network outage at 1-Aug right after 6 hours of UDP storm from 07:xx-12:xx, it was reasonable to assume that these UDP packets were generated due to equipment faults. Futher investigation may need to capture the UDP content as well as the ARP addresses of the equipment that generated these packets.

spp_http_decode: CGI Null Byte attack detected

This is snort's spp_http_decode preprocessor to check for %00 string (null byte) in http packets. It only accounted for 2.39% of the total alerts.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	3	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
01	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
02	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
05	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	16	0.00%	40	0.00%	27083	1.21%	0	0.00%	0	0.00%
09	3744	0.17%	75	0.00%	0	0.00%	0	0.00%	0	0.00%
10	0	0.00%	143	0.01%	0	0.00%	1991	0.09%	3524	0.16%
11	234	0.01%	14	0.00%	0	0.00%	0	0.00%	22	0.00%
12	1	0.00%	0	0.00%	30	0.00%	0	0.00%	236	0.01%
13	0	0.00%	1165	0.05%	0	0.00%	38	0.00%	7525	0.34%
14	0	0.00%	6	0.00%	0	0.00%	57	0.00%	3586	0.16%
15	0	0.00%	300	0.01%	0	0.00%	0	0.00%	3610	0.16%
16	0	0.00%	0	0.00%	0	0.00%	0	0.00%	84	0.00%
17	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
18	8	0.00%	9	0.00%	0	0.00%	0	0.00%	0	0.00%
19	0	0.00%	0	0.00%	0	0.00%	0	0.00%	1	0.00%
20	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
21	0	0.00%	0	0.00%	0	0.00%	0	0.00%	7	0.00%
22	0	0.00%	7	0.00%	0	0.00%	0	0.00%	6	0.00%
23	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

Table 10. Distribution of CGI Null Byte attack alerts in 5-day period

From the distribution above, we can identify mainly six major incidents of this alert.

The alert at 1-Aug 09:xx was mainly triggered by internal host MY.NET.70.14 target against 216.241.219.28. Some were triggered by another internal host MY.NET.145.160 mainly against some hosts of hp.com.

Alerts	Source IP	Destination IP	Remarks
3578	MY.NET.70.48	216.241.219.28	Strange IP, no dns, no record in whois of nic.com.
120	MY.NET.145.160	62.4.73.68	62.4.73.68.speedera.cdg2.fr.mfnx.net
24	MY.NET.145.160	192.151.52.111	atwnt370.external.hp.com
10	MY.NET.145.160	192.151.52.22	us-support2.external.hp.com
8	MY.NET.145.160	192.6.234.10	tux414-f.q-tam.hp.com
3	MY.NET.145.160	192.151.10.31	pawnt172.external.hp.com
1	MY.NET.145.160	192.6.164.81	dux412.den.hp.com

Table 11. CGI Null Byte alerts generated by MY.NET.70.48 & MY.NET.145.160

To my surprise, I could not find any information about the target IP 216.241.219.28. No dns reverse lookup, no record in nic.com and dshield.org. But there really existed an web server at 216.241.219.28 that did not give a meaningful web page. This alert was rather suspicious. However, both MY.NET.70.48 and 216.241.219.28 did not appear in other alerts. Without the payload of the packets, no further interpretation could be made.

The alert at 2-Aug at 13:xx was mainly triggered by IP MY.NET.85.78.

Alerts	Source IP	Destination IP	Remarks
1164	MY.NET.85.78	209.10.239.135	no dns, no record at nic.com, but 131 records in dshield.org against this IP
1	MY.NET.87.98	205.158.62.91	205-158-62-91.outblaze.com

Table 12. CGI Null Byte alerts generated by MY.NET.85.78 & MY.NET.87.98

The same situation happened again for the target IP 209.10.239.135. No dns reverse lookup, no record at nic.com and do not give a meaningful web page when loaded. But there were records against this IP address in dshield.org. IP 209.10.239.135 do not engaged in other alerts but MY.NET.85.78 showed signs of Trojan.

```
8/05-18:53:02.268091  [**] Possible trojan server activity [**] 63.196.247.234:3403 -> MY.NET.85.78:27374
08/05-18:53:02.320302  [**] Possible trojan server activity [**] 63.196.247.234:3440 -> MY.NET.85.78:27374
08/05-18:53:05.129004  [**] Possible trojan server activity [**] 63.196.247.234:3403 -> MY.NET.85.78:27374
```

The third incident appeared at 3-Aug 08:xx. All alerts were triggered by MY.NET.81.37 against 216.241.219.28.

Alerts	Source IP	Destination IP	Remarks
27083	MY.NET.81.37	216.241.219.28	The same target IP as the incident at 1-Aug

Table 13. CGI Null Byte alerts generated by MY.NET.81.37

MY.NET.81.37 did not show signs of Trojan but there were two other IIS Unicode alerts associated with it.

```
08/02-00:14:55.813686  [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.81.37:4725 ->
198.172.122.10:80
08/02-00:20:46.048834  [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.81.37:4745 ->
198.172.122.10:80
```

For the alert at 4-Aug 10:xx, alerts were targetted against 209.10.239.135 again, but this time from another internal host MY.NET.84.189.

Alerts	Source IP	Destination IP	Remarks
1991	MY.NET.84.189	209.10.239.135	The same target IP as the incident at 2-Aug

Table 14. CGI Null Byte attack alerts generated by MY.NET.84.189

The internal host MY.NET.84.189 showed serious signs of Trojan.

```
08/02-09:05:07.810900  [**] Possible trojan server activity [**] 80.62.155.240:3825 -> MY.NET.84.189:27374
08/02-09:05:07.811045  [**] Possible trojan server activity [**] MY.NET.84.189:27374 -> 80.62.155.240:3825
08/02-09:05:09.024532  [**] Possible trojan server activity [**] 80.62.155.240:3825 -> MY.NET.84.189:27374
08/02-09:05:09.024550  [**] Possible trojan server activity [**] MY.NET.84.189:27374 -> 80.62.155.240:3825
08/04-09:22:08.503321  [**] Possible trojan server activity [**] 61.102.149.115:3901 -> MY.NET.84.189:27374
08/04-09:22:08.503489  [**] Possible trojan server activity [**] MY.NET.84.189:27374 -> 61.102.149.115:3901
08/04-09:22:09.173735  [**] Possible trojan server activity [**] 61.102.149.115:3901 -> MY.NET.84.189:27374
08/04-09:22:09.173990  [**] Possible trojan server activity [**] MY.NET.84.189:27374 -> 61.102.149.115:3901
08/04-09:22:09.873923  [**] Possible trojan server activity [**] 61.102.149.115:3901 -> MY.NET.84.189:27374
08/04-09:22:09.874063  [**] Possible trojan server activity [**] MY.NET.84.189:27374 -> 61.102.149.115:3901
```

The alerts at 5-Aug 10:xx, another internal host MY.NET.87.52 involved.

Alerts	Source IP	Destination IP	Remarks
3523	MY.NET.87.52	216.241.219.28	Suspect hacker IP at 1-Aug and 3-Aug
1	MY.NET.87.98	205.158.62.90	205-158-62-91.outblaze.com

Table 15. CGI Null Byte attack alerts generated by MY.NET.87.52 & MY.NET.87.98

The IP MY.NET.87.52 also involved in three minor alerts.

```
08/05-16:32:28.098220  [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.87.52:55850 -
> 66.135.209.133:80
08/05-16:32:28.178293  [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.87.52:55850 -
> 66.135.209.133:80
08/05-16:32:28.220225  [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.87.52:55850 -
> 66.135.209.133:80
```

The serious incident occurred at 5-Aug 13:xx-15:xx. There were total 6 internal hosts involved against 17 different target hosts.

Alerts	Source IP	Destination IP	Remarks
5085	MY.NET.178.219 (no other alerts)	216.241.219.28	Suspect hacker IP at 1-Aug and 3-Aug & 5-Aug 10:xx
3979	MY.NET.182.91 (IIS Unicode attack as well)	152.163.210.84	sand-int-v11.ptn.aol.com dshield.org: 3 records against this IP to 1 target host
1180		63.121.98.69	63-121-98-69.focaldata.net dshield.org: 72 records against this IP to 4 targets
140		207.189.75.40	ironport01.mlb.com dshield.org: 38 records against this IP to 16 targets

80		63.121.106.133	digital-island.133.focaldata.net dshield.org: 2126 records against this IP to 36 hosts
2160	MY.NET.109.83 (IIS Unicode attack as well)	152.163.210.84	sand-int-v11.ptn.aol.com dshield.org: 3 records against this IP to 1 target
1710		63.208.106.67	unknown.Level3.net dshield.org: 99 records against this IP to 18 targets
288		63.121.98.69	63-121-98-69.focaldata.net dshield.org: 72 records against this IP to 4 targets
18		63.121.106.133	digital-island.133.focaldata.net dshield.org: 2126 records against this IP to 36 hosts
43	MY.NET.87.52 (Possible myserver activity alert)	66.135.192.224	pics.ebay.com dshield.org: 797 records against this IP to 112 targets
13		66.135.192.229	include.ebay.com dshield.org: 70 records against this IP to 25 targets
4		216.32.120.133	pages.ebay.com dshield.org: 627 records against this IP to 141 targets
4		66.135.193.137	thumbs.ebay.com dshield.org: 583 records against this IP to 134 targets
3		66.135.194.135	search.ebay.com dshield.org: 7 records against this IP to 7 targets
1		216.32.120.142	listings.ebay.com dshield.org: 57 records against 10 hosts
1		216.32.120.145	members.ebay.com dshield.org: no match
4	MY.NET.153.206 (IIS Unicode attack as well)	216.33.88.141	www.cars.com dshield.org:1 record against 1 host
4		128.167.120.48	no dns dshield.org: 105 records against 55 targets
2		205.138.3.82	no dns dshield.org: 99 records against 60 targets
1	MY.NET.90.59 (IIS Unicode attack as well)	66.128.224.102	bor-clust12.ofoto.com dshield.org: 1 record against 1 host

Table 16. CGI Null Byte attack alerts at 5-Aug 13:xx-15:xx

From the above table, one can easily see that most of those destination IP addresses were "bad guys". It was not difficult to infer that those internal hosts involved were either compromised or controlled by hackers. If this was really the case, the hacker(s) behind the scene may be the type that were professional and sophisticated, as there were a number of hosts from well known sites (e.g. ebay.com) under their control. All we need was the packet payload to determine what was really involved in these alerts.

Recommendations: Administrator of the University follow up with the internal hosts listed in Table11-16.

SMB Name Wildcard

This alert account for only 1.34% of the total alerts. When a Windows box is trying to find out name for a particular IP address, it will first perform a DNS lookup. If a negative response comes back, it will query Netbios. When resolving with Netbios, it will send out a UDP "NodeStauts" query, that is sent from the Windows machine port 137 to the target machine's port 137. A possible snort rule may be something like this:

```
alert udp any any -> $HOME_NET 137 (msg:"SMB Name Wildcard";
content:"CKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA|0000|";)
```

Here is the distribution table.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	195	0.01%	138	0.01%	135	0.01%	373	0.02%	216	0.01%
01	188	0.01%	122	0.01%	138	0.01%	639	0.03%	229	0.01%
02	228	0.01%	149	0.01%	94	0.00%	277	0.01%	326	0.01%
03	155	0.01%	155	0.01%	82	0.00%	467	0.02%	193	0.01%
04	154	0.01%	280	0.01%	67	0.00%	416	0.02%	151	0.01%
05	194	0.01%	143	0.01%	85	0.00%	151	0.01%	79	0.00%
06	254	0.01%	174	0.01%	187	0.01%	200	0.01%	279	0.01%
07	370	0.02%	158	0.01%	609	0.03%	607	0.03%	287	0.01%
08	229	0.01%	236	0.01%	872	0.04%	592	0.03%	271	0.01%
09	241	0.01%	146	0.01%	320	0.01%	806	0.04%	121	0.01%
10	415	0.02%	79	0.00%	296	0.01%	524	0.02%	234	0.01%
11	251	0.01%	153	0.01%	165	0.01%	411	0.02%	196	0.01%
12	207	0.01%	171	0.01%	115	0.01%	259	0.01%	170	0.01%
13	0	0.00%	171	0.01%	163	0.01%	402	0.02%	104	0.00%
14	0	0.00%	216	0.01%	145	0.01%	314	0.01%	112	0.00%
15	0	0.00%	278	0.01%	158	0.01%	143	0.01%	137	0.01%
16	0	0.00%	276	0.01%	200	0.01%	155	0.01%	170	0.01%
17	0	0.00%	169	0.01%	1015	0.05%	200	0.01%	200	0.01%
18	135	0.01%	103	0.00%	1277	0.06%	151	0.01%	372	0.02%
19	97	0.00%	375	0.02%	402	0.02%	160	0.01%	371	0.02%
20	139	0.01%	527	0.02%	113	0.01%	167	0.01%	279	0.01%
21	98	0.00%	267	0.01%	222	0.01%	235	0.01%	237	0.01%
22	199	0.01%	211	0.01%	197	0.01%	313	0.01%	535	0.02%
23	210	0.01%	204	0.01%	215	0.01%	381	0.02%	355	0.02%

Table 17. Distribution of SMB Name Wildcard alerts in 5-day period

From the distribution table, one can easily see that this is an extremely noisy alerts. A statistics on the log files indicated that there were 8957 unique source hosts and there were 2742 target hosts. Actually no meaningful analysis can be done on such a noisy alert.

TFTP - External UDP connection to internal tftp server

This was again, another noisy alert which account for 1.08% of total alerts. TFTP is a simple protocol used to transfer files using a UDP connection. The rule likely appears as:

```
alert udp $INTERNAL 69 -> any any (msg:"TFTP - External UDP connection
to internal tftp server";)
```

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	134	0.01%	228	0.01%	110	0.00%	272	0.01%	256	0.01%
01	179	0.01%	171	0.01%	168	0.01%	272	0.01%	328	0.01%
02	142	0.01%	217	0.01%	116	0.01%	237	0.01%	275	0.01%
03	169	0.01%	193	0.01%	117	0.01%	193	0.01%	249	0.01%
04	123	0.01%	343	0.02%	100	0.00%	244	0.01%	165	0.01%
05	101	0.00%	206	0.01%	103	0.00%	192	0.01%	253	0.01%
06	203	0.01%	210	0.01%	101	0.00%	292	0.01%	274	0.01%
07	178	0.01%	206	0.01%	302	0.01%	309	0.01%	269	0.01%
08	209	0.01%	213	0.01%	317	0.01%	353	0.02%	246	0.01%
09	160	0.01%	151	0.01%	337	0.02%	320	0.01%	198	0.01%
10	188	0.01%	97	0.00%	222	0.01%	377	0.02%	161	0.01%
11	89	0.00%	102	0.00%	235	0.01%	294	0.01%	151	0.01%
12	100	0.00%	120	0.01%	185	0.01%	338	0.02%	130	0.01%
13	0	0.00%	131	0.01%	181	0.01%	305	0.01%	152	0.01%
14	0	0.00%	88	0.00%	188	0.01%	334	0.01%	74	0.00%
15	0	0.00%	96	0.00%	216	0.01%	290	0.01%	138	0.01%
16	0	0.00%	130	0.01%	205	0.01%	238	0.01%	117	0.01%
17	0	0.00%	121	0.01%	330	0.01%	135	0.01%	245	0.01%
18	118	0.01%	102	0.00%	368	0.02%	203	0.01%	162	0.01%
19	170	0.01%	204	0.01%	309	0.01%	146	0.01%	190	0.01%
20	167	0.01%	222	0.01%	191	0.01%	271	0.01%	215	0.01%
21	183	0.01%	283	0.01%	318	0.01%	381	0.02%	222	0.01%
22	226	0.01%	207	0.01%	378	0.02%	243	0.01%	226	0.01%
23	247	0.01%	208	0.01%	306	0.01%	358	0.02%	192	0.01%

Table 18. Distribution of TFTP - External UDP connection to internal tftp server alerts in 5-day period

Most of these alerts were generated by 4 internal hosts continuously.

Alerts	Source IP	Destination IP	Remarks
6090	MY.NET.111.230	192.168.0.216	MY.NET.111.230 no Nimda alert detected
6059	MY.NET.111.231	192.168.0.216	MY.NET.111.231 no Nimda alert detected
6053	MY.NET.109.105	192.168.0.216	MY.NET.109.105 no Nimda alert detected
6007	MY.NET.111.219	192.168.0.216	MY.NET.111.219 no Nimda alert detected
2	199.106.211.166	MY.NET.117.25	199.106.211.166 Red Worm alert detected as well

2	209.61.187.112	MY.NET.180.39	209.61.187.112 Red Worm alert detected as well
1	MY.NET.109.105	MY.NET.104.204	
1	63.250.205.12	MY.NET.114.44	

Table 19. Internal hosts that generated TFTP alerts

By looking at the log entries, we can easily see that the internal hosts were sending from port 69 to a high port of 192.168.0.216. These look like replies from tftp server to a high port client. However, we don't see alerts triggered by client requests.

```
08/01-03:55:37.559544  [**] TFTP - External UDP connection to internal tftp server [**] MY.NET.111.230:69 ->
192.168.0.216:9920
08/01-03:55:41.575747  [**] TFTP - External UDP connection to internal tftp server [**] MY.NET.111.230:69 ->
192.168.0.216:9920
08/01-03:55:49.623247  [**] TFTP - External UDP connection to internal tftp server [**] MY.NET.111.230:69 ->
192.168.0.216:8776
08/01-03:55:52.385451  [**] TFTP - External UDP connection to internal tftp server [**] MY.NET.111.230:69 ->
192.168.0.216:6513
```

One may think that the snort rule may not capturing alerts from the other direction. However, this was the not case since we did had a log that an external IP 63.250.205.12 as a client, made a tftp connection to port 69 of an internal host MY.NET.114.44. Look at this.

```
08/05-13:34:24.979258  [**] TFTP - External UDP connection to internal tftp server [**] 63.250.205.12:40961 ->
MY.NET.114.44:69
```

The best explaination to this was that the above alerts was created due to some non-properly configured firewall. May be there were internal hosts configured with private IP address 192.168.0.216. NAT (Network Address Translation) was not done properly and tftp connections were made to MY.NET.111.x tftp servers. When these tftp servers replied, they replied to the private address 192.168.0.216 and packets were routed to the Internet (due to default route for unknown addresses) and then picked up by the IDS host near border routers. So, these alerts were, actually, false postive alarms.

Recommendations: Administrator of the University should check the internal hosts in Table 19. Determine why tftp servers were (or if) running. Any virus infection? Try to locate the host that use the IP 192.168.0.216 and fix and address translation problem.

External RPC call

This alert accounted for 0.65% of total alerts. It was at first quite difficult to determine what was the content of the snort rule that generated this alert. In standard package of snort, there were different rules to detect different RPC function calls. The alerts here also did not tell wether the traffic were TCP or UDP. However, by comparing log entries in the scan logs and alerts, one can tell that the rule involved here was simply detect an external host making a TCP connection to an internal host's port 111. Here is an extract from scans.logs.

```
Aug  1 07:45:10 203.239.155.2:45601 -> MY.NET.80.40:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45602 -> MY.NET.80.41:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45603 -> MY.NET.80.42:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45604 -> MY.NET.80.43:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45605 -> MY.NET.80.44:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45608 -> MY.NET.80.47:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45606 -> MY.NET.80.45:111 SYN ******S*
Aug  1 07:45:10 203.239.155.2:45609 -> MY.NET.80.48:111 SYN ******S*
```

Here is the corresponding alert logs.

```
08/01-07:45:10.982730  [**] External RPC call [**] 203.239.155.2:45601 -> MY.NET.80.40:111
08/01-07:45:10.982858  [**] External RPC call [**] 203.239.155.2:45602 -> MY.NET.80.41:111
08/01-07:45:10.983121  [**] External RPC call [**] 203.239.155.2:45603 -> MY.NET.80.42:111
08/01-07:45:10.983133  [**] External RPC call [**] 203.239.155.2:45604 -> MY.NET.80.43:111
08/01-07:45:10.983142  [**] External RPC call [**] 203.239.155.2:45605 -> MY.NET.80.44:111
08/01-07:45:10.983996  [**] External RPC call [**] 203.239.155.2:45608 -> MY.NET.80.47:111
08/01-07:45:10.984039  [**] External RPC call [**] 203.239.155.2:45606 -> MY.NET.80.45:111
08/01-07:45:10.984205  [**] External RPC call [**] 203.239.155.2:45609 -> MY.NET.80.48:111
```

Thus, the snort rule generated these alerts may be look like this"

```
alert TCP $EXTERNAL any -> $INTERNAL 111 (msg: "External RPC call";
flags: S+;)
```

Knowing what the alert was detecting, we can now look at the distribution table.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	11	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
01	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
02	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
05	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	918	0.04%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
09	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
10	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
11	0	0.00%	0	0.00%	0	0.00%	918	0.04%	0	0.00%
12	0	0.00%	0	0.00%	0	0.00%	3601	0.16%	0	0.00%
13	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
14	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
15	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
16	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
17	0	0.00%	3	0.00%	8352	0.37%	0	0.00%	0	0.00%
18	149	0.01%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
19	627	0.03%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
20	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
21	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
22	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
23	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

Table 20. Distribution of Exernal RPC call alerts in 5-day period

From the distribution table, it was clear that there were 4 distinct incidents involving this alert.

At 1-Aug 07:xx, there was a port scan activity from 203.239.155.2 to port 111 of 793 internal hosts. At 1-Aug 18:xx-19:xx, another host 202.108.109.100 performed similar port scan to port 111 of 294 internal hosts. The suspect IP 203.239.155.2 also found in another alert but 202.108.109.100 did not.

```
08/01-07:47:25.051127  [**] STATDX UDP attack [**] 203.239.155.2:937 -> MY.NET.104.116:1024
08/01-07:47:48.541475  [**] STATDX UDP attack [**] 203.239.155.2:948 -> MY.NET.136.3:32802
08/01-07:47:49.301140  [**] STATDX UDP attack [**] 203.239.155.2:949 -> MY.NET.139.161:862
08/01-07:47:50.091741  [**] STATDX UDP attack [**] 203.239.155.2:950 -> MY.NET.139.163:805
08/01-07:47:51.505009  [**] STATDX UDP attack [**] 203.239.155.2:952 -> MY.NET.139.200:798
```

Woo! A seach on dshield.org for IP 203.239.155.2 give 39980 records against this IP to 37991 targets!! Searching for IP 202.108.109.100 on dshield.org also yielded 95705 records against the IP to 60949 targets.

In another incident occurred at 3-Aug 17:xx, an external IP 194.98.189.139 scanned port 111 of 5456 internal hosts. This was a rather aggressive port scanning activity. This IP also involed in another alert, much like 203.239.155.2 above, suggested these may be activities of the same hacker.

```
08/03-17:26:17.360497  [**] STATDX UDP attack [**] 194.98.189.139:837 -> MY.NET.5.31:32772
08/03-17:31:04.051081  [**] STATDX UDP attack [**] 194.98.189.139:877 -> MY.NET.110.70:32778
08/03-17:31:05.107995  [**] STATDX UDP attack [**] 194.98.189.139:880 -> MY.NET.110.86:32780
08/03-17:31:21.681730  [**] STATDX UDP attack [**] 194.98.189.139:883 -> MY.NET.130.42:32773
08/03-17:31:36.308785  [**] STATDX UDP attack [**] 194.98.189.139:886 -> MY.NET.136.3:32803
08/03-17:31:38.479837  [**] STATDX UDP attack [**] 194.98.189.139:887 -> MY.NET.139.15:32768
08/03-17:31:39.581480  [**] STATDX UDP attack [**] 194.98.189.139:891 -> MY.NET.139.49:32768
08/03-17:31:39.936251  [**] STATDX UDP attack [**] 194.98.189.139:892 -> MY.NET.139.50:32768
```

A search for the IP 194.98.189.139 on dshield.org yielded 391 records against the IP to 379 targets.

In the last incident at 4-Aug 11:xx-12:xx, an external IP 61.182.50.241 scanned port 111 of 3518 internal hosts. This IP also involved in another alert "STATDX UDP attack". Again, the same type of attack as the previous two.

```
08/04-11:55:29.393230  [**] STATDX UDP attack [**] 61.182.50.241:678 -> MY.NET.1.2:1024
08/04-11:57:01.187940  [**] STATDX UDP attack [**] 61.182.50.241:696 -> MY.NET.27.3:32774
08/04-12:01:17.301181  [**] STATDX UDP attack [**] 61.182.50.241:759 -> MY.NET.136.3:32803
08/04-12:01:21.701083  [**] STATDX UDP attack [**] 61.182.50.241:764 -> MY.NET.139.25:32769
08/04-12:01:24.039772  [**] STATDX UDP attack [**] 61.182.50.241:766 -> MY.NET.139.40:32769
08/04-12:01:26.043685  [**] STATDX UDP attack [**] 61.182.50.241:766 -> MY.NET.139.40:32769
```

A search of dsheild.org indicated that there were 87114 records against this IP to 69568 targets. Attack to 111 from this IP was especially highlighted in the search result.

From the timing of the port scan and attack, it was quite clear that this was some kind of script kiddies. Internal hosts with port 111 open were determined first and then an STATDX attack was attempted. Administrator of the University should check those hosts that were attacked by "STATDX UDP attack" to see if any host was successfully compromised.

Watchlist 000220 IL-ISDNNET-990517

This alert accounted for only 0.53% of the total alerts. This seemed to be a alert triggered by a special snort rule made by the University to monitor activities from or to the network 212.179.0.0/17. A search for the keyword "IL-ISDNNET-990517" on google come up with pages pointed to sans.org that reported hacking activities from that network. Probably this is a rule setup to monitor these hacking activities. The snort rule may be something like this.

```
alert TCP 212.179.0.0/17 any -> $INTERNAL any (msg: "Watchlist 000220 IL-
ISDNNET-990517"; flags: A+;)
```

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	4	0.00%	10	0.00%	0	0.00%	6	0.00%	0	0.00%
01	20	0.00%	0	0.00%	4	0.00%	6	0.00%	9	0.00%
02	71	0.00%	0	0.00%	7	0.00%	0	0.00%	6	0.00%
03	44	0.00%	3	0.00%	55	0.00%	6	0.00%	0	0.00%
04	34	0.00%	14	0.00%	0	0.00%	38	0.00%	0	0.00%
05	44	0.00%	107	0.00%	0	0.00%	46	0.00%	6	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	44	0.00%
07	0	0.00%	0	0.00%	0	0.00%	15	0.00%	9	0.00%
08	24	0.00%	81	0.00%	22	0.00%	59	0.00%	106	0.00%
09	1673	0.07%	54	0.00%	1500	0.07%	68	0.00%	6	0.00%
10	796	0.04%	11	0.00%	4	0.00%	72	0.00%	16	0.00%
11	152	0.01%	21	0.00%	6	0.00%	130	0.01%	10	0.00%
12	217	0.01%	97	0.00%	19	0.00%	58	0.00%	9	0.00%
13	0	0.00%	123	0.01%	13	0.00%	92	0.00%	36	0.00%
14	0	0.00%	1799	0.08%	1500	0.07%	60	0.00%	3	0.00%
15	0	0.00%	488	0.02%	49	0.00%	27	0.00%	55	0.00%
16	0	0.00%	97	0.00%	1172	0.05%	2	0.00%	11	0.00%
17	0	0.00%	45	0.00%	85	0.00%	12	0.00%	1	0.00%
18	6	0.00%	11	0.00%	21	0.00%	19	0.00%	13	0.00%
19	0	0.00%	1	0.00%	5	0.00%	23	0.00%	6	0.00%
20	6	0.00%	2	0.00%	0	0.00%	0	0.00%	1	0.00%
21	7	0.00%	21	0.00%	6	0.00%	0	0.00%	243	0.01%
22	0	0.00%	5	0.00%	16	0.00%	0	0.00%	0	0.00%
23	8	0.00%	6	0.00%	7	0.00%	1	0.00%	0	0.00%

Table 21. Distribution of Watchlist 000220 IL-ISDNNET-990517 alerts in 5-day period

From the distribution table above, we noticed that this was indeed a noisy alert. Yet we can still identify 5 incidents with distinctive activities from ISDNNET.

The first incident at 1-Aug 09:xx was due to activities from 212.179.99.182 to internal host MY.NET.108.42 (1535 alerts out of total 1673). A look at the logs indicated 212.179.99.182 were making connections to port 1214 of MY.NET.108.42.

```
08/01-09:46:37.756937  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.99.182:1565 ->
MY.NET.108.42:1214
08/01-09:46:38.364750  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.99.182:1577 ->
MY.NET.108.42:1214
08/01-09:46:38.671572  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.99.182:1577 ->
MY.NET.108.42:1214
08/01-09:46:38.686169  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.99.182:1565 ->
MY.NET.108.42:1214
```

Port 1214 was used by a P2P file sharing program called "Kazaa". It depended on the University's policy whether users were allowed to use this kind of Peer-to-Peer program.

The incident at 2-Aug 14:xx was mainly due to activities of some ISDNNET hosts to two internal host MY.NET.153.196 (1275 alerts) and MY.NET.153.153 (486 alerts). This alerts were caused by web browsing activities from these two internal hosts to some web servers on ISDNNET. No suspicious activity this time.

```
08/02-14:20:13.104106  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.196:1300
08/02-14:20:13.106295  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.196:1300
08/02-14:20:13.115274  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.22:80 ->
MY.NET.153.196:1304
08/02-14:20:13.115283  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.22:80 ->
MY.NET.153.196:1304
```

The incident at 3-Aug 09:xx was caused by activities from two ISDNNET host 212.179.83.22 to internal host MY.NET.104.204 (1070 alerts) and ISDNNET host 212.179.83.8 to internal host MY.NET.117.137 (418 alerts) respectively. The were all Kazaa activities.

```
08/03-09:03:36.602453  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.86.5:4431 ->
MY.NET.104.204:1214
08/03-09:03:39.908501  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.86.5:4431 ->
MY.NET.104.204:1214
…skipped
08/03-09:04:02.415709  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.83.8:3321 ->
MY.NET.117.137:1214
08/03-09:04:03.111477  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.83.8:3321 ->
MY.NET.117.137:1214
```

The incident at 3-Aug 14:xx was mainly due to activities from ISDNNET host 212.179.66.17 to internal host MY.NET.153.196 (1468 alerts). This was web browsing activities from internal host to ISDNNET host.

```
08/03-14:47:58.209165  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.196:2332
08/03-14:47:58.220591  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.196:2332
```

The last incidet at 3-Aug 16:xx was due to activities from ISDNNET hosts to internal host MY.NET.153.159 (1172 alerts). These alerts were caused by web browsing activites from internal host to ISDNNET hosts.

```
08/03-16:34:42.937590  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.159:2676
08/03-16:34:42.944060  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.159:2677
08/03-16:34:42.944134  [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.66.17:80 ->
MY.NET.153.159:2677
```

Possible trojan server activity

This alert accounted for 0.18% of total alerts. These alerts seemed to be triggered by rules that monitor the usage of port 27374. Port 27374 was used by a Trojan called Sub-seven. When Sub-seven was implanted in a victim system, can grant remote hacker complete control of the system. The snort rules for detect the Trojan may look like this:

```
alert TCP any any -> $INTERNAL 27374 (msg: "Possible trojan server
activity"; flags: A+;)
alert TCP $INTERNAL 27374 -> any any (msg: "Possible trojan server
activity"; flags: A+;)
```

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	6	0.00%	22	0.00%	0	0.00%	0	0.00%	0	0.00%
01	3	0.00%	0	0.00%	1	0.00%	0	0.00%	1	0.00%
02	3	0.00%	0	0.00%	0	0.00%	0	0.00%	16	0.00%
03	0	0.00%	0	0.00%	0	0.00%	2	0.00%	8	0.00%
04	0	0.00%	2	0.00%	0	0.00%	0	0.00%	0	0.00%
05	2	0.00%	0	0.00%	0	0.00%	0	0.00%	96	0.00%
06	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	0	0.00%	0	0.00%	1	0.00%	0	0.00%	0	0.00%
09	2	0.00%	619	0.03%	6	0.00%	602	0.03%	4	0.00%
10	2	0.00%	0	0.00%	2	0.00%	0	0.00%	0	0.00%
11	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
12	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
13	0	0.00%	0	0.00%	0	0.00%	55	0.00%	0	0.00%
14	0	0.00%	0	0.00%	0	0.00%	558	0.02%	2	0.00%
15	0	0.00%	0	0.00%	0	0.00%	0	0.00%	2	0.00%
16	0	0.00%	0	0.00%	0	0.00%	0	0.00%	2	0.00%
17	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
18	0	0.00%	0	0.00%	0	0.00%	0	0.00%	1109	0.05%
19	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
20	192	0.01%	0	0.00%	1	0.00%	0	0.00%	6	0.00%
21	0	0.00%	0	0.00%	2	0.00%	0	0.00%	0	0.00%
22	0	0.00%	3	0.00%	0	0.00%	0	0.00%	0	0.00%
23	1	0.00%	123	0.01%	658	0.03%	0	0.00%	2	0.00%

Table 22. Distribution Possible trojan server activity alerts in 5-day period.

From the distribution table, there were 5 distinctive incidents. The incident at 2-Aug 09:xx was due to activities of external host 80.62.155.240. A search in the scan logs for this IP indicated that 80.62.155.240 was performing port scan activities and searching for Trojaned hosts at that time.

```
Aug  2 09:04:56 80.62.155.240:2550 -> MY.NET.84.155:27374 SYN ******S*
Aug  2 09:04:56 80.62.155.240:2569 -> MY.NET.84.155:12345 SYN ******S*
Aug  2 09:04:56 80.62.155.240:2591 -> MY.NET.84.156:27374 SYN ******S*
Aug  2 09:04:57 80.62.155.240:2610 -> MY.NET.84.156:12345 SYN ******S*
Aug  2 09:04:56 80.62.155.240:2631 -> MY.NET.84.157:27374 SYN ******S*
Aug  2 09:04:56 80.62.155.240:2650 -> MY.NET.84.157:12345 SYN ******S*
Aug  2 09:04:57 80.62.155.240:2671 -> MY.NET.84.158:27374 SYN ******S*
Aug  2 09:04:57 80.62.155.240:2692 -> MY.NET.84.158:12345 SYN ******S*
Aug  2 09:04:56 80.62.155.240:2711 -> MY.NET.84.159:27374 SYN ******S*
Aug  2 09:04:58 80.62.155.240:2730 -> MY.NET.84.159:12345 SYN ******S*
```

Some of the internal hosts responded to the port scan and triggered these alerts. There were 152 unique internal hosts that responded positively to the port scan. Administrator of the University may need to have a check on these internal hosts.

MY.NET.83.124	MY.NET.83.30	MY.NET.84.184	MY.NET.84.250
MY.NET.83.126	MY.NET.83.33	MY.NET.84.186	MY.NET.84.252
MY.NET.83.131	MY.NET.83.38	MY.NET.84.189	MY.NET.84.254
MY.NET.83.133	MY.NET.83.54	MY.NET.84.190	MY.NET.84.26
MY.NET.83.146	MY.NET.83.56	MY.NET.84.191	MY.NET.84.27
MY.NET.83.147	MY.NET.83.6	MY.NET.84.192	MY.NET.84.30
MY.NET.83.150	MY.NET.83.72	MY.NET.84.193	MY.NET.85.101
MY.NET.83.159	MY.NET.83.82	MY.NET.84.195	MY.NET.85.114
MY.NET.83.160	MY.NET.83.95	MY.NET.84.196	MY.NET.85.115
MY.NET.83.163	MY.NET.84.12	MY.NET.84.201	MY.NET.85.119
MY.NET.83.165	MY.NET.84.130	MY.NET.84.202	MY.NET.85.24
MY.NET.83.167	MY.NET.84.131	MY.NET.84.203	MY.NET.85.25
MY.NET.83.177	MY.NET.84.132	MY.NET.84.204	MY.NET.85.35
MY.NET.83.179	MY.NET.84.134	MY.NET.84.205	MY.NET.85.42
MY.NET.83.189	MY.NET.84.136	MY.NET.84.206	MY.NET.85.51
MY.NET.83.19	MY.NET.84.138	MY.NET.84.207	MY.NET.85.52
MY.NET.83.190	MY.NET.84.144	MY.NET.84.208	MY.NET.85.53
MY.NET.83.196	MY.NET.84.147	MY.NET.84.21	MY.NET.85.58
MY.NET.83.197	MY.NET.84.148	MY.NET.84.210	MY.NET.85.59
MY.NET.83.201	MY.NET.84.150	MY.NET.84.213	MY.NET.85.60
MY.NET.83.202	MY.NET.84.155	MY.NET.84.216	MY.NET.85.66
MY.NET.83.206	MY.NET.84.156	MY.NET.84.217	MY.NET.85.68
MY.NET.83.208	MY.NET.84.158	MY.NET.84.222	MY.NET.85.74
MY.NET.83.209	MY.NET.84.159	MY.NET.84.223	MY.NET.85.75
MY.NET.83.212	MY.NET.84.160	MY.NET.84.224	MY.NET.85.87
MY.NET.83.213	MY.NET.84.162	MY.NET.84.226	MY.NET.85.88
MY.NET.83.214	MY.NET.84.164	MY.NET.84.227	MY.NET.85.91
MY.NET.83.215	MY.NET.84.165	MY.NET.84.228	MY.NET.85.92
MY.NET.83.220	MY.NET.84.166	MY.NET.84.229	MY.NET.85.96
MY.NET.83.221	MY.NET.84.167	MY.NET.84.23	MY.NET.85.97
MY.NET.83.223	MY.NET.84.168	MY.NET.84.233	MY.NET.85.98

MY.NET.83.224	MY.NET.84.169	MY.NET.84.234	MY.NET.85.99
MY.NET.83.226	MY.NET.84.170	MY.NET.84.235	
MY.NET.83.227	MY.NET.84.171	MY.NET.84.236	
MY.NET.83.232	MY.NET.84.172	MY.NET.84.237	
MY.NET.83.235	MY.NET.84.174	MY.NET.84.239	
MY.NET.83.24	MY.NET.84.175	MY.NET.84.244	
MY.NET.83.247	MY.NET.84.178	MY.NET.84.246	
MY.NET.83.25	MY.NET.84.179	MY.NET.84.247	
MY.NET.83.251	MY.NET.84.183	MY.NET.84.25	

Table 23. Internal hosts that responded to trojan scan

Incident at 3-Aug 23:xx was due to another IP 218.154.202.148. It also performed similar port scan activities at that time and a number of internal hosts responded.

```
Aug  3 23:02:48 218.154.202.148:1230 -> MY.NET.152.11:27374 SYN ******S*
Aug  3 23:02:48 218.154.202.148:1248 -> MY.NET.152.11:12345 SYN ******S*
Aug  3 23:02:48 218.154.202.148:1271 -> MY.NET.152.12:27374 SYN ******S*
Aug  3 23:02:48 218.154.202.148:1277 -> MY.NET.152.12:12345 SYN ******S*
Aug  3 23:02:48 218.154.202.148:1293 -> MY.NET.152.13:27374 SYN ******S*
Aug  3 23:02:49 218.154.202.148:1319 -> MY.NET.152.13:12345 SYN ******S*
Aug  3 23:02:49 218.154.202.148:1347 -> MY.NET.152.14:27374 SYN ******S*
Aug  3 23:02:49 218.154.202.148:1351 -> MY.NET.152.14:12345 SYN ******S*
```

The scanning was performed on another segment and there were 51 unique internal
hosts responded positively to the port scan.

MY.NET.152.11	MY.NET.152.22	MY.NET.153.182	MY.NET.153.197
MY.NET.152.12	MY.NET.153.163	MY.NET.153.184	MY.NET.153.198
MY.NET.152.126	MY.NET.153.164	MY.NET.153.185	MY.NET.153.202
MY.NET.152.13	MY.NET.153.165	MY.NET.153.186	MY.NET.153.203
MY.NET.152.14	MY.NET.153.168	MY.NET.153.187	MY.NET.153.205
MY.NET.152.15	MY.NET.153.170	MY.NET.153.188	MY.NET.153.206
MY.NET.152.17	MY.NET.153.171	MY.NET.153.189	MY.NET.153.208
MY.NET.152.19	MY.NET.153.172	MY.NET.153.190	MY.NET.153.209
MY.NET.152.21	MY.NET.153.173	MY.NET.153.191	MY.NET.153.210
MY.NET.152.213	MY.NET.153.174	MY.NET.153.193	MY.NET.153.211
MY.NET.152.214	MY.NET.153.175	MY.NET.153.194	MY.NET.153.219
MY.NET.152.215	MY.NET.153.176	MY.NET.153.195	MY.NET.153.71
MY.NET.152.216	MY.NET.153.179	MY.NET.153.196	

Table 24. Internal hosts that responded to trojan scan

Another incident at 4-Aug 09:xx, was caused by scanning activity from IP 61.102.149.115.

```
Aug  4 09:21:58 61.102.149.115:3309 -> MY.NET.84.160:27374 SYN ******S*
Aug  4 09:21:58 61.102.149.115:3318 -> MY.NET.84.160:12345 SYN ******S*
Aug  4 09:21:59 61.102.149.115:3342 -> MY.NET.84.162:27374 SYN ******S*
Aug  4 09:21:59 61.102.149.115:3348 -> MY.NET.84.162:12345 SYN ******S*
Aug  4 09:21:58 61.102.149.115:3382 -> MY.NET.84.163:27374 SYN ******S*
Aug  4 09:21:58 61.102.149.115:3403 -> MY.NET.84.163:12345 SYN ******S*
Aug  4 09:21:59 61.102.149.115:3407 -> MY.NET.84.164:27374 SYN ******S*
Aug  4 09:21:59 61.102.149.115:3418 -> MY.NET.84.164:12345 SYN ******S*
Aug  4 09:22:00 61.102.149.115:3431 -> MY.NET.84.165:27374 SYN ******S*
Aug  4 09:22:00 61.102.149.115:3441 -> MY.NET.84.165:12345 SYN ******S*
```

The target segments were the same as the scanning activity detected at 2-Aug 09:xx but this time only 99 unique hosts responded.

At 14:xx the same day, another IP 217.136.63.141 port scan again.

```
Aug  4 14:02:16 217.136.63.141:3867 -> MY.NET.84.160:27374 SYN ******S*
Aug  4 14:02:16 217.136.63.141:3933 -> MY.NET.84.160:12345 SYN ******S*
Aug  4 14:02:16 217.136.63.141:4199 -> MY.NET.84.162:12345 SYN ******S*
Aug  4 14:02:16 217.136.63.141:4152 -> MY.NET.84.162:27374 SYN ******S*
```

The target segments the same as that in the morning and 98 unique hosts responded, more or less the same result in the morning.

The fifth incident occurred at 5-Aug 18:xx. The alerts were due to scanning activities of 63.196.247.234. Again, the target segments were the same at 2-Aug and 4-Aug.

```
Aug  5 18:52:06 63.196.247.234:4307 -> MY.NET.84.131:27374 SYN ******S*
Aug  5 18:52:06 63.196.247.234:4430 -> MY.NET.84.131:12345 SYN ******S*
Aug  5 18:52:07 63.196.247.234:4855 -> MY.NET.84.134:27374 SYN ******S*
Aug  5 18:52:07 63.196.247.234:4928 -> MY.NET.84.134:12345 SYN ******S*
Aug  5 18:52:07 63.196.247.234:1065 -> MY.NET.84.135:12345 SYN ******S*
Aug  5 18:52:07 63.196.247.234:1026 -> MY.NET.84.135:27374 SYN ******S*
```

There were 135 internal hosts responded and the list were more or less the same as that of 2-Aug.

Recommendations: Administrator of the University were advised to check those internal hosts listed in Table 23 and Table 24.

SUNRPC highport access!

This alert only accounted for 0.11% of total alerts. Normally, the rpcbind service only listens on port 111. Under Solaris, the rpcbind service also listens under port 32771, which sometimes allows attackers to bypass packet filtering.

The snort rule detecting this alert may be like this:

```
alert tcp any any -> $INTERNAL 32771 (msg: "SUNRPC highport access!";)
```

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
01	0	0.00%	0	0.00%	0	0.00%	17	0.00%	0	0.00%
02	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	3	0.00%	0	0.00%	2426	0.11%	0	0.00%	0	0.00%
05	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	0	0.00%	0	0.00%	0	0.00%	48	0.00%	0	0.00%
07	0	0.00%	6	0.00%	0	0.00%	0	0.00%	0	0.00%
08	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
09	0	0.00%	0	0.00%	0	0.00%	0	0.00%	13	0.00%
10	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
11	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
12	0	0.00%	4	0.00%	0	0.00%	0	0.00%	0	0.00%
13	0	0.00%	0	0.00%	0	0.00%	0	0.00%	6	0.00%
14	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
15	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
16	0	0.00%	1	0.00%	4	0.00%	0	0.00%	0	0.00%
17	0	0.00%	2	0.00%	0	0.00%	0	0.00%	0	0.00%
18	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
19	0	0.00%	7	0.00%	0	0.00%	0	0.00%	0	0.00%
20	5	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
21	1	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
22	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
23	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

Table 25. Distribution of SUNRPC highport access alerts in 5-day period

There were only 1 incident that caught our attention. The incident at 3-Aug 04:xx was caused by external host 140.90.198.34 against internal host MY.NET.154.27. This was a very aggressive attack as there were 290 alerts generated per second!!

```
08/03-04:57:05.473097  [**] SUNRPC highport access! [**] 140.90.198.134:8289 -> MY.NET.154.27:32771
08/03-04:57:05.475186  [**] SUNRPC highport access! [**] 140.90.198.134:8289 -> MY.NET.154.27:32771
08/03-04:57:05.475975  [**] SUNRPC highport access! [**] 140.90.198.134:8289 -> MY.NET.154.27:32771
08/03-04:57:05.476124  [**] SUNRPC highport access! [**] 140.90.198.134:8289 -> MY.NET.154.27:32771
08/03-04:57:05.476750  [**] SUNRPC highport access! [**] 140.90.198.134:8289 -> MY.NET.154.27:32771
…skipped
```

Recommendations: We don't know whether this aggressive attack lead to a successful hack or not. We did not find any suspicious hacking activities initiated from MY.NET.154.27. Anyway, administrator of the University was advised to have a check on this host as well.

Top Alert Speakers

Actually we have came across all these alert speakers in the previous discussion of alerts. Here is a table summaring them again.

Alerts	Source IP Address	Remarks
1444025	MY.NET.100.208	Host infected by Nimda at 5-Aug 20:xx, triggered many Nimda attack & IIS Unicode attack
483715	MY.NET.84.234	Port scan alerts at 4-Aug 17:xx - 19:xx, triggered "IIS Overflow ida" alerts at the same time
51405	3.0.0.99	Triggered UDP alerts all over the time. UDP connection from 3.0.0.99:137 to 10.0.0.1:137
32161	63.250.213.12	Triggered UDP alerts at 1-Aug from 63.250.213.12:1031 to 233.28.65.148:5779
27085	MY.NET.81.37	Triggered CGI Null Byte alerts at 3-Aug 08:xx against 216.241.219.28
8375	194.98.189.139	At 3-Aug 17:xx, scanned port 111 of 5456 internal hosts
6994	MY.NET.85.74	Triggered IIS Unicode alerts, target against 207.200.86.97 from 1-Aug to 5-Aug, target against 64.12.147.25 at 5-Aug
6905	80.137.90.34	Triggered IIS Unicode alerts at 5-Aug 10:00. Target against MY.NET.5.14. Stopped after generating 75min traffic.
6092	MY.NET.111.230	TFTP alerts with target 192.168.0.216
6062	MY.NET.111.231	TFTP alerts with target 192.168.0.216

Table 26. Top Alert Speakers

Top Scan Speakers

Here is the distribution table for scanning after removing those UDP entries from the log files.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	315	0.03%	75	0.01%	476	0.05%	114	0.01%	10309	1.03%
01	311	0.03%	264	0.03%	1177	0.12%	1107	0.11%	10557	1.06%
02	75	0.01%	43	0.00%	2985	0.30%	3223	0.32%	14849	1.49%
03	2019	0.20%	65	0.01%	4402	0.44%	156	0.02%	13099	1.31%
04	5206	0.52%	578	0.06%	15	0.00%	5996	0.60%	85	0.01%
05	10123	1.01%	406	0.04%	11	0.00%	3963	0.40%	46	0.00%
06	1137	0.11%	9307	0.93%	13	0.00%	7892	0.79%	334	0.03%
07	1672	0.17%	4316	0.43%	3356	0.34%	5096	0.51%	679	0.07%
08	323	0.03%	153	0.02%	7684	0.77%	5423	0.54%	192	0.02%
09	397	0.04%	257	0.03%	145	0.01%	4865	0.49%	1553	0.16%
10	194	0.02%	508	0.05%	59	0.01%	1806	0.18%	15996	1.60%
11	1929	0.19%	1001	0.10%	30	0.00%	16796	1.68%	7847	0.79%
12	2319	0.23%	3595	0.36%	440	0.04%	6601	0.66%	140	0.01%
13	0	0.00%	3976	0.40%	4216	0.42%	4571	0.46%	104	0.01%
14	0	0.00%	75	0.01%	147	0.01%	1641	0.16%	90	0.01%
15	0	0.00%	105	0.01%	680	0.07%	18235	1.83%	672	0.07%
16	0	0.00%	309	0.03%	274	0.03%	1778	0.18%	1232	0.12%
17	0	0.00%	5088	0.51%	7889	0.79%	111055	11.13%	6826	0.68%
18	9018	0.90%	1795	0.18%	7797	0.78%	251342	25.19%	481	0.05%
19	4922	0.49%	161	0.02%	120	0.01%	137547	13.79%	2975	0.30%
20	4597	0.46%	295	0.03%	96	0.01%	441	0.04%	5162	0.52%
21	62	0.01%	644	0.06%	247	0.02%	370	0.04%	49286	4.94%
22	103	0.01%	953	0.10%	77	0.01%	111	0.01%	75996	7.62%
23	47	0.00%	157	0.02%	813	0.08%	14238	1.43%	52803	5.29%

Table 27. Distribution of scanning activities in 5-day period

Scans	%	Source IP Address	Time of incident	No. of unique hosts scanned	Port scanned	dshield.org search	Other Remarks
478410	47.95%	MY.NET.84.234	4-Aug 17:xx - 19:xx	478374	80	no match	Triggered "IIS Overflow ida" alerts at the same time
170142	17.05%	MY.NET.100.208	5-Aug 21:xx - 23:xx	100392	80	no match	Host infected by Nimda at 5-Aug 20:xx
25015	2.51%	216.228.171.81	1-Aug 06:xx - 4-Aug 13:xx	5307	139 & 445	no match	Scan constantly across the time period, also triggered "SMB Name Wildcard" alerts
21019	2.11%	24.138.61.171	4-Aug 17:xx-19:xx	6757	80	1 records, 1 host	No other alerts
20329	2.04%	161.132.205.100	5-Aug 02:xx-03:xx	7176	80	5 records, 3 hosts	No other alerts except beetle.ucs
17730	1.78%	211.232.192.153	4-Aug 10:xx-12:xx	7276	1433	816 records, 369 hosts	No other alerts except beetle.ucs
16263	1.63%	67.104.84.142	5-Aug 01:xx-02:xx	7194	1433	410 records, 333 hosts	No other alerts
15741	1.58%	219.96.171.20	4-Aug 23:xx	7147	80	no match	No other alerts
15693	1.57%	80.137.90.34	5-Aug 10:xx-11:xx	6579	80	no match	No other alerts
12593	1.26%	24.101.152.5	4-Aug 14:xx-16:xx	5884	21	4 records, 3 targets	No other alerts

Table 28. Top 10 Scan Speakers

Top OOS Speakers

There were generally three reasons for Out of Spec packets:
1. Packet corruption
2. Implementation of Explicit Congestion Notification standard (ECN in RFC2481)
3. Crafted packets designed for portscanning and OS fingerprinting.

Hour of day	1-Aug		2-Aug		3-Aug		4-Aug		5-Aug	
00	117	7.15%	0	0.00%	2	0.12%	0	0.00%	0	0.00%
01	209	12.77%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
02	74	4.52%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
03	105	6.41%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
04	97	5.93%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
05	72	4.40%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
06	58	3.54%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
07	67	4.09%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
08	68	4.15%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
09	143	8.74%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
10	80	4.89%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
11	34	2.08%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
12	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
13	0	0.00%	5	0.31%	0	0.00%	0	0.00%	0	0.00%
14	0	0.00%	54	3.30%	0	0.00%	0	0.00%	0	0.00%
15	0	0.00%	22	1.34%	0	0.00%	0	0.00%	0	0.00%
16	0	0.00%	57	3.48%	0	0.00%	0	0.00%	0	0.00%
17	0	0.00%	37	2.26%	0	0.00%	0	0.00%	0	0.00%
18	0	0.00%	69	4.22%	0	0.00%	0	0.00%	0	0.00%
19	0	0.00%	48	2.93%	0	0.00%	0	0.00%	0	0.00%
20	0	0.00%	54	3.30%	0	0.00%	0	0.00%	0	0.00%
21	0	0.00%	49	2.99%	0	0.00%	1	0.06%	0	0.00%
22	0	0.00%	46	2.81%	0	0.00%	0	0.00%	0	0.00%
23	0	0.00%	62	3.79%	0	0.00%	0	0.00%	0	0.00%

Table 29. Distribution of OOS in 5-day period

From the table above, it was quite clear that there were two main incidents of OOS. One occurred at 1-Aug 00:xx-11:xx. Another incident occurred at 2-Aug 13:xx-23:xx. However, it was difficult to differentiate whether these incidents were caused by corrupted packets or malicious crafted packets. Table 30 listed the top OOS source IP addresses. Within those IP addresses only 209.116.70.75 and 209.132.232.101 seemed malicious. But they only accounted for 14.17% of total OOS activities.

On the other hand, ECN standard calls for setting the two reserved flags in the TCP header on the initial SYN to perform a network congestion check. However, this was not widely supported by vendor yet. If ECN was involved in the above OOS activities, there should not be only two distinct incidents in the 5-day period.

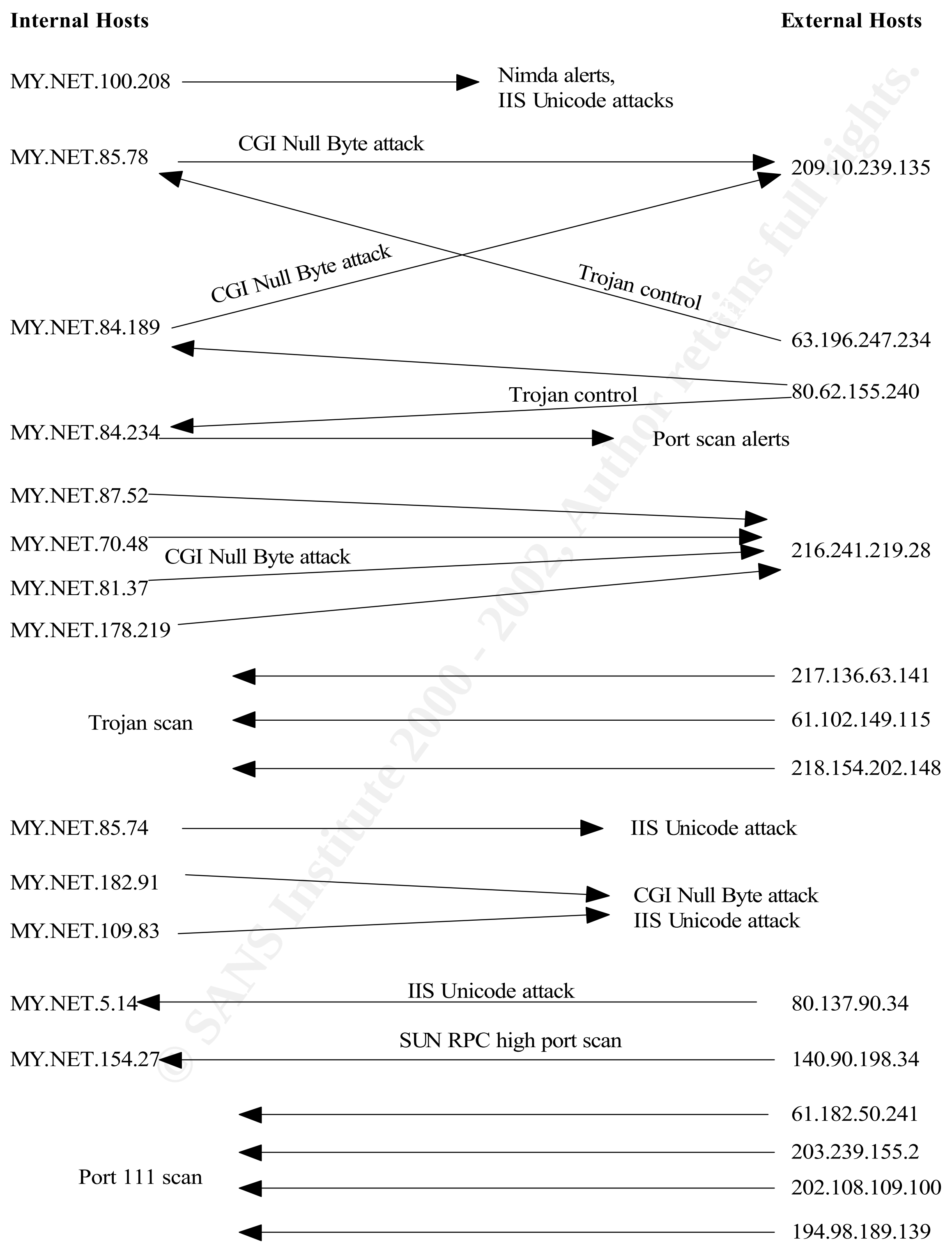
To my best guess, majority of these OOS activities should be caused by packet corruptions. May be some network equipments in the campus were not functioning in good health. The network outage in 1-Aug 13:xx-17:xx seemed support this point.

Scans	%	Source IP Address	Time of incident	No. of unique hosts scanned	Port scanned	dshield. org search	Other Remarks
652	39.83%	68.32.126.64	1-Aug 00:xx-11:xx, 2-Aug 13:xx-23:xx	1	110	2 records, 1 target	target MY.NET.6.7 only, no other scan & alert
345	21.08%	62.76.241.129	1-Aug 00:xx-02:xx, 1-Aug 09:xx-10:xx	2	113	no match	target MY.NET.97.217 & MY.NET.97.238 only, no other scan & alert
214	13.07%	209.116.70.75	1-Aug 00:xx-11:xx, 2-Aug 13:xx-23:xx	8	25	53 records, 7 targets	triggered over 600 "Queso fingerprint" alerts, scans with reserved bits
83	5.07%	212.35.180.17	1-Aug 02:xx-11:xx	1	21	no mtach	target MY.NET.253.20 only, no other scan & alert
48	2.93%	65.210.154.210	1-Aug 00:xx-05:xx, 2-Aug 15:xx-23:xx	1	4662	no match	target MY.NET.111.198 only
29	1.77%	213.250.44.19	1-Aug 03:xx	2	80	no match	target MY.NET.253.114 & MY.NET.253.125 only
18	1.10%	61.132.74.239	1-Aug 04:xx, 06:xx	1	80	1 record, 1 target	target MY.NET.100.165 only
18	1.10%	209.132.232.101	1-Aug 10:xx-11:xx, 2-Aug 14:xx,20:xx-22:xx	7	25	117 records, 19 targets	triggered "Queso fingerprint" alerts, scans with reserve bits
18	1.10%	202.155.91.142	1-Aug 04:xx-05:xx	2	80	no match	target MY.NET.253.125 & MY.NET.60.14 only
17	1.04%	211.154.85.159	1-Aug 03:xx-06:xx	1	80	no match	target MY.NET.111.140 only

Table 30. Top OOS Source

Link Graph Analysis

The Link Graph below summarized the most significant IP addresses involved in the alerts within the 5-day period.



One thing that caught my attention was that, three internal host MY.NET.85.78, MY.NET.84.189 and MY.NET.84.234 seemed to be controlled by hackers to launch attack against others. Other four internal hosts MY.NET.87.52, MY.NET.70.48, MY.NET.81.37 and MY.NET.178,.219 were also sources of attack to a particular external host 216.241.219.28.

Registrant Information of External IP Addresses Involved in Link Graph

```
==================================================
61.102.149.115 - Torjan scan

<net block>
61.96.0.0 - 61.111.255.255

<owner>
KRNIC
Korea Network Information Center
South Korea

<administrative contact>
Host Master
11F, KTF B/D, 1321-11, Seocho2-Dong, Seocho-Gu,
Seoul, Korea, 137-857
phone: +82-2-2186-4500
fax: +82-2-2186-4496
hostmaster@nic.or.kr

<technical contact>
Host Master
11F, KTF B/D, 1321-11, Seocho2-Dong, Seocho-Gu,
Seoul, Korea, 137-857
phone: +82-2-2186-4500
fax: +82-2-2186-4496
hostmaster@nic.or.kr

<additional data>
KRNIC-KR
Updated: 06-Jun-2001 by hostmaster@apnic.net
Source: whois.apnic.net

==================================================
61.182.50.241 - Port 111 scan

<net block>
61.182.0.0 - 61.182.255.255

<owner>
CHINANET Hebei province network
Data Communication Division
China Telecom

<administrative contact>
Dongmei Kou
A12,Xin-Jie-Kou-Wai Street,
Beijing,100088
phone: +86-10-62370437
fax: +86-10-62053995
chunguangcanlanxiaobajie@sina.com

<technical contact>
zhiyong chen
hebei province shijiazhuang
fanxi road No.19
hebei shuju tongxin ju
phone: +86-311-6051394
fax: +86-311-6672895
jixin@sj-user.he.cninfo.net
```

```
<additional data>
CHINANET-HE
Updated: 16-Feb-2001 by hostmaster@ns.chinanet.cn.net
Source: whois.apnic.net

==================================================
```
63.196.247.234 - Trojan control
```
adsl-63-196-247-234.dsl.lsan03.pacbell.net

<net block>
63.196.240.0 - 63.196.247.255

<owner>
PPPoX Pool Rback7 63.196.240.0
268 Bush St #5000 San Francisco CA 94104
United States

<technical contact>
IPAdmin-PBI
+1-888-212-5411
IPAdmin-PBI@sbis.sbc.com

<additional data>
SBCIS-062002142653
Created: 2002-06-21
Updated: 2002-06-21
Source: whois.arin.net

==================================================
```
80.62.155.240 - Torjan control
```
0x503e9bf0.odnxx4.adsl-dhcp.tele.dk

<net block>
80.62.155.0 - 80.62.155.255

<owner>
IP addresses for ADSL users in
Tele Danmark\'s IP backbone.
Location: Odense
Box: odnxx4
Denmark

<administrative contact>
AS3292 Staff
Tele Danmark DataNet
Sletvej 30, A039
DK-8310 Tranbjerg
Denmark
phone: +45 50 12 29 47
staff@ip.tele.dk

<technical contact>
AS3292 Staff
Tele Danmark DataNet
Sletvej 30, A039
DK-8310 Tranbjerg
Denmark
phone: +45 50 12 29 47
staff@ip.tele.dk

<additional data>
TDC-TELEDANMARK-BREDBAANDSADSL-NET
Updated: 11-Jun-2001 by heves@tdk.dk
Source: whois.ripe.net

==================================================
```
80.137.90.34 - IIS Unicode attack
```
p50895A22.dip.t-dialin.net

<net block>
80.128.0.0 - 80.146.159.255
```

```
<owner>
Deutsche Telekom AG
Germany

<administrative contact>
DTAG Global IP-Adressing
Deutsche Telekom AG
Bayreuther Strasse 1
D-90409 Nuernberg
Germany
phone: +49 911 68909856
ripe.dtip@telekom.de

<technical contact>
Security Team
Deutsche Telekom AG
Technikniederlassung Schwaebisch Hall
D-89070 Ulm
Germany
phone: +49 731 100 84055
fax: +49 731 100 84150
abuse@t-ipnet.de

<additional data>
DTAG-DIAL16
Updated: 08-Jan-2002 by auftrag@nic.telekom.de
Source: whois.ripe.net

==================================================
```

140.90.198.34 - SUN RPC high port scan

```
<net block>
140.90.0.0 - 140.90.255.255

<owner>
National Oceanic and Atmospheric Administration
1315 East-West Highway Silver Spring MD 20910
United States

<technical contact>
Kyler, John
+1-301-713-0600
John.C.Kyler@noaa.gov

<name servers>
NEWNS.NOAA.GOV
NWRNS.NOAA.GOV
SERNS.NOAA.GOV
MERNS.NOAA.GOV

<additional data>
NOAA-NET
Created: 1990-04-09
Updated: 2002-01-09
Source: whois.arin.net

==================================================
```

194.98.189.139 - Port 111 scan

```
<net block>
194.98.189.128 - 194.98.189.143

<owner>
INGENCYS
France

<administrative contact>
Monsieur De Royer
INGENCYS
4, Rue de la Madeleine
45140 ST JEAN DE LA RUELLE, France
phone: +33 2 37 25 12 00
```

```
fax: +33 2 37 25 12 00

<technical contact>
technical contact
UUNET FRANCE
215, Avenue Georges Clemenceau
F-92024 NANTERRE Cedex
phone: +33 1 56 38 22 00
fax: +33 1 56 38 22 01
net-adm@mciworldcom.fr

<additional data>
INGENCYS-NET1
Updated: 24-Sep-2001 by frederic.martzel@mciworldcom.fr
Source: whois.ripe.net

==================================================
```

202.108.109.100 - Port 111 scan

```
<net block>
202.108.109.0 - 202.108.109.255

<owner>
Beijing Guang Xinwang Digital
Technology Co.Ltd
China

<administrative contact>
He JianBo
Dong Zhong Jie 9 Dong Cheng District
Beijing 100027
phone: +86-10-64181150-215
fax: +86-10-64181819
jumper@btamail.net.cn

<technical contact>
He JianBo
Dong Zhong Jie 9 Dong Cheng District
Beijing 100027
phone: +86-10-64181150-215
fax: +86-10-64181819
jumper@btamail.net.cn

<additional data>
BJ-GX-DIGIT-TECH-CO
Updated: 16-Apr-2002 by suny@publicf.bta.net.cn
Source: whois.apnic.net

==================================================
```

203.239.155.2 - Port 111 scan

```
<net block>
203.232.0.0 - 203.239.255.255

<owner>
KRNIC
Korea Network Information Center
South Korea

<administrative contact>
Host Master
11F, KTF B/D, 1321-11, Seocho2-Dong, Seocho-Gu,
Seoul, Korea, 137-857
phone: +82-2-2186-4500
fax: +82-2-2186-4496
hostmaster@nic.or.kr

<technical contact>
Host Master
11F, KTF B/D, 1321-11, Seocho2-Dong, Seocho-Gu,
Seoul, Korea, 137-857
phone: +82-2-2186-4500
```

```
fax: +82-2-2186-4496
hostmaster@nic.or.kr

<additional data>
KRNIC-KR
Updated: 06-Jun-2001 by hostmaster@apnic.net
Source: whois.apnic.net

==================================================
```
209.10.239.135 - Attacked by CGI Null Byte

```
<net block>
209.10.239.128 - 209.10.239.191

<owner>
IFilm Corp
1024 North Orange Drive HOLLYWOOD CA 90038
United States

<technical contact>
Swipper
+1-212-625-7777
swipper@globix.net
Hostmaster, Globix
+1-212-334-8500
arin-admin@globix.net

<additional data>
IP007442-209-10-239
Created: 2002-10-01
Updated: 2002-10-01
Source: whois.arin.net

==================================================
```
216.241.219.28 - Attacked by CGI Null Byte

```
<net block>
216.241.208.0 - 216.241.223.255

<owner>
The Cobalt Group, Inc
2030 1st Avenue Seattle WA 98121
United States

<technical contact>
Fitzgerald, Michael
+1-800-909-8244
mikef@cobaltgroup.com

<additional data>
COBALT-NET2
Created: 1999-11-16
Updated: 1999-11-16
Source: whois.arin.net

==================================================
```
217.136.63.141 - Trojan scan

```
141.63-136-217.adsl.skynet.be

<net block>
217.136.48.0 - 217.136.63.255

<owner>
Belgacom Skynet SA/NV
ADSL BAS Antwerpen TL GO/PLUS
Belgium

<administrative contact>
Skynet NOC administrators
Belgacom Skynet  SA/NV
rue colonel Bourg 124
B-1140 Brussels
```

```
Belgium
phone: +3227061311
fax: +3227269311

<technical contact>
Skynet NOC administrators
Belgacom Skynet  SA/NV
rue colonel Bourg 124
B-1140 Brussels
Belgium
phone: +3227061311
fax: +3227269311

<additional data>
BE-SKYNET-20010125
Updated: 03-Feb-2001 by piet@skynet.be
Source: whois.ripe.net
```

```
==================================================
```

218.154.202.148 - Trojan scan

```
<net block>
218.154.202.0 - 218.154.202.255

<owner>
PUSAN NODE
75 4KA JUNGANGDONG JUNGKU
PUSAN
600-711
South Korea

<administrative contact>
GilSoon Park
phone: +82-2-747-9213
fax: +82-2-766-5901
gspark@kornet.net

<technical contact>
Won Kang
phone: +82-2-747-9213
fax: +82-2-766-5901
ip@ns.kornet.net

<additional data>
KORNET-XDSL-PUSAN-KR
Updated: 23-Sep-2002 by hostmaster@nic.or.kr
Source: whois.apnic.net

==================================================
```

Conclusions and Defensive Recommendations

The University's security issues can be summarized as:

- Nimda and IIS worm infection
- Machines compromised and controlled by Trojan horses to launch external attacks
- Serious port scan activities from external hackers
- Abnormal traffic may be caused by systems not probably configured or systems with malfunctioned hardware

To prevent worm infections, the most important thing is to have all servers with the most updated security patches applied. For client users, to have the most updated virus signature. To prevent loosely configured systems to plugged into public network, policy needed to be enforced when implementing Internet systems. Guidelines of configuring system and applying patches should be provided by the Security party of the University and users are encouraged to follow.

To enhace security of client machines so as to avoid worms and Trojans, centralized server for updating virus signatures can be used. Virus signatures on client machines can be updated automatically when the user login. Awareness training also needed to provide to users, so that users know how and where to report when there is a suspected security issue.

The University may need to consider implementing a border firewall system that filter unauthorized port scan activities. The firewall can be used to filter vulnerabile services like SUNRPC and Trojan.

To Minimize abnormal traffic, ingress and egress filtering can be enabled on routers within the University campus. Traffic monitoring system like MRTG can be implemented to monitoring traffic pattern.

The security party of the University also need to assess the security of the campus network periodically. Security scans can be performed and discovered any system vulnerabilities before the system be compromised. Up-to-date security news also needed to maintain host security.

Tools Used for "Analyze This!"

All logs were uploaded to a Linux server for analysis. All analysis were done by using shell and awk scripts.

First, all alert logs were concatenated together to a large file called "alert_all" and all "spp_portscan" lines were removed. To generate Table 1 used in this analysis, the following script was run. The result was import into Excel to make up the table.

```
#!/bin/bash
echo -n "08/01-00";grep "08/01-00" alert_all | wc -l
echo -n "08/01-01";grep "08/01-01" alert_all | wc -l
echo -n "08/01-02";grep "08/01-02" alert_all | wc -l
echo -n "08/01-03";grep "08/01-03" alert_all | wc -l
echo -n "08/01-04";grep "08/01-04" alert_all | wc -l
echo -n "08/01-05";grep "08/01-05" alert_all | wc -l
echo -n "08/01-06";grep "08/01-06" alert_all | wc -l
echo -n "08/01-07";grep "08/01-07" alert_all | wc -l
echo -n "08/01-08";grep "08/01-08" alert_all | wc -l
echo -n "08/01-09";grep "08/01-09" alert_all | wc -l
echo -n "08/01-10";grep "08/01-10" alert_all | wc -l
echo -n "08/01-11";grep "08/01-11" alert_all | wc -l
echo -n "08/01-12";grep "08/01-12" alert_all | wc -l
echo -n "08/01-13";grep "08/01-13" alert_all | wc -l
echo -n "08/01-14";grep "08/01-14" alert_all | wc -l
echo -n "08/01-15";grep "08/01-15" alert_all | wc -l
echo -n "08/01-16";grep "08/01-16" alert_all | wc -l
echo -n "08/01-17";grep "08/01-17" alert_all | wc -l
echo -n "08/01-18";grep "08/01-18" alert_all | wc -l
echo -n "08/01-19";grep "08/01-19" alert_all | wc -l
echo -n "08/01-20";grep "08/01-20" alert_all | wc -l
echo -n "08/01-21";grep "08/01-21" alert_all | wc -l
echo -n "08/01-22";grep "08/01-22" alert_all | wc -l
echo -n "08/01-23";grep "08/01-23" alert_all | wc -l
echo -n "08/02-00";grep "08/02-00" alert_all | wc -l
echo -n "08/02-01";grep "08/02-01" alert_all | wc -l
echo -n "08/02-02";grep "08/02-02" alert_all | wc -l
echo -n "08/02-03";grep "08/02-03" alert_all | wc -l
echo -n "08/02-04";grep "08/02-04" alert_all | wc -l
echo -n "08/02-05";grep "08/02-05" alert_all | wc -l
echo -n "08/02-06";grep "08/02-06" alert_all | wc -l
echo -n "08/02-07";grep "08/02-07" alert_all | wc -l
echo -n "08/02-08";grep "08/02-08" alert_all | wc -l
echo -n "08/02-09";grep "08/02-09" alert_all | wc -l
echo -n "08/02-10";grep "08/02-10" alert_all | wc -l
echo -n "08/02-11";grep "08/02-11" alert_all | wc -l
echo -n "08/02-12";grep "08/02-12" alert_all | wc -l
echo -n "08/02-13";grep "08/02-13" alert_all | wc -l
echo -n "08/02-14";grep "08/02-14" alert_all | wc -l
echo -n "08/02-15";grep "08/02-15" alert_all | wc -l
echo -n "08/02-16";grep "08/02-16" alert_all | wc -l
echo -n "08/02-17";grep "08/02-17" alert_all | wc -l
echo -n "08/02-18";grep "08/02-18" alert_all | wc -l
echo -n "08/02-19";grep "08/02-19" alert_all | wc -l
echo -n "08/02-20";grep "08/02-20" alert_all | wc -l
echo -n "08/02-21";grep "08/02-21" alert_all | wc -l
echo -n "08/02-22";grep "08/02-22" alert_all | wc -l
echo -n "08/02-23";grep "08/02-23" alert_all | wc -l
echo -n "08/03-00";grep "08/03-00" alert_all | wc -l
echo -n "08/03-01";grep "08/03-01" alert_all | wc -l
echo -n "08/03-02";grep "08/03-02" alert_all | wc -l
echo -n "08/03-03";grep "08/03-03" alert_all | wc -l
echo -n "08/03-04";grep "08/03-04" alert_all | wc -l
echo -n "08/03-05";grep "08/03-05" alert_all | wc -l
echo -n "08/03-06";grep "08/03-06" alert_all | wc -l
echo -n "08/03-07";grep "08/03-07" alert_all | wc -l
echo -n "08/03-08";grep "08/03-08" alert_all | wc -l
echo -n "08/03-09";grep "08/03-09" alert_all | wc -l
echo -n "08/03-10";grep "08/03-10" alert_all | wc -l
echo -n "08/03-11";grep "08/03-11" alert_all | wc -l
echo -n "08/03-12";grep "08/03-12" alert_all | wc -l
echo -n "08/03-13";grep "08/03-13" alert_all | wc -l
echo -n "08/03-14";grep "08/03-14" alert_all | wc -l
echo -n "08/03-15";grep "08/03-15" alert_all | wc -l
echo -n "08/03-16";grep "08/03-16" alert_all | wc -l
echo -n "08/03-17";grep "08/03-17" alert_all | wc -l
echo -n "08/03-18";grep "08/03-18" alert_all | wc -l
echo -n "08/03-19";grep "08/03-19" alert_all | wc -l
echo -n "08/03-20";grep "08/03-20" alert_all | wc -l
echo -n "08/03-21";grep "08/03-21" alert_all | wc -l
echo -n "08/03-22";grep "08/03-22" alert_all | wc -l
echo -n "08/03-23";grep "08/03-23" alert_all | wc -l
echo -n "08/04-00";grep "08/04-00" alert_all | wc -l
echo -n "08/04-01";grep "08/04-01" alert_all | wc -l
echo -n "08/04-02";grep "08/04-02" alert_all | wc -l
echo -n "08/04-03";grep "08/04-03" alert_all | wc -l
echo -n "08/04-04";grep "08/04-04" alert_all | wc -l
echo -n "08/04-05";grep "08/04-05" alert_all | wc -l
echo -n "08/04-06";grep "08/04-06" alert_all | wc -l
echo -n "08/04-07";grep "08/04-07" alert_all | wc -l
echo -n "08/04-08";grep "08/04-08" alert_all | wc -l
echo -n "08/04-09";grep "08/04-09" alert_all | wc -l
echo -n "08/04-10";grep "08/04-10" alert_all | wc -l
echo -n "08/04-11";grep "08/04-11" alert_all | wc -l
echo -n "08/04-12";grep "08/04-12" alert_all | wc -l
echo -n "08/04-13";grep "08/04-13" alert_all | wc -l
echo -n "08/04-14";grep "08/04-14" alert_all | wc -l
echo -n "08/04-15";grep "08/04-15" alert_all | wc -l
echo -n "08/04-16";grep "08/04-16" alert_all | wc -l
echo -n "08/04-17";grep "08/04-17" alert_all | wc -l
echo -n "08/04-18";grep "08/04-18" alert_all | wc -l
echo -n "08/04-19";grep "08/04-19" alert_all | wc -l
echo -n "08/04-20";grep "08/04-20" alert_all | wc -l
echo -n "08/04-21";grep "08/04-21" alert_all | wc -l
echo -n "08/04-22";grep "08/04-22" alert_all | wc -l
echo -n "08/04-23";grep "08/04-23" alert_all | wc -l
echo -n "08/05-00";grep "08/05-00" alert_all | wc -l
echo -n "08/05-01";grep "08/05-01" alert_all | wc -l
echo -n "08/05-02";grep "08/05-02" alert_all | wc -l
echo -n "08/05-03";grep "08/05-03" alert_all | wc -l
echo -n "08/05-04";grep "08/05-04" alert_all | wc -l
echo -n "08/05-05";grep "08/05-05" alert_all | wc -l
echo -n "08/05-06";grep "08/05-06" alert_all | wc -l
echo -n "08/05-07";grep "08/05-07" alert_all | wc -l
echo -n "08/05-08";grep "08/05-08" alert_all | wc -l
echo -n "08/05-09";grep "08/05-09" alert_all | wc -l
echo -n "08/05-10";grep "08/05-10" alert_all | wc -l
echo -n "08/05-11";grep "08/05-11" alert_all | wc -l
echo -n "08/05-12";grep "08/05-12" alert_all | wc -l
echo -n "08/05-13";grep "08/05-13" alert_all | wc -l
echo -n "08/05-14";grep "08/05-14" alert_all | wc -l
echo -n "08/05-15";grep "08/05-15" alert_all | wc -l
echo -n "08/05-16";grep "08/05-16" alert_all | wc -l
echo -n "08/05-17";grep "08/05-17" alert_all | wc -l
echo -n "08/05-18";grep "08/05-18" alert_all | wc -l
echo -n "08/05-19";grep "08/05-19" alert_all | wc -l
echo -n "08/05-20";grep "08/05-20" alert_all | wc -l
echo -n "08/05-21";grep "08/05-21" alert_all | wc -l
echo -n "08/05-22";grep "08/05-22" alert_all | wc -l
echo -n "08/05-23";grep "08/05-23" alert_all | wc -l
```

To analyze a particular alert, first the alerts were extracted to a file first. For example,

```
grep "SMB Name Wildcard" alert_all > smb_wild.txt
```

To generate a list of hosts that triggered this alert in decending order:

```
#!/bin/bash
cat smb_wild.txt | awk '{print $9}' | awk -F: '{print $1}' | sort |
count.awk
```

Here count.awk was a awk script that count the occurrence of an IP address and output the occurrence and the IP address together.

```
#!/usr/bin/awk -f

# This script count the occurence of first column (pre-sorted)
NR == 1 {prev=$1;n=0}
$1 != prev {printf "%4d %s\n",n,prev;prev=$1;n=0}
$1 == prev {n++}
END {printf "%4d %s\n",n,prev}
```

Then to generate a distribution table, a script like the follow was run:

```
#!/bin/bash
echo -n "08/01-00";grep "08/01-00" smb_wild.txt | wc -l
echo -n "08/01-01";grep "08/01-01" smb_wild.txt | wc -l
echo -n "08/01-02";grep "08/01-02" smb_wild.txt | wc -l
echo -n "08/01-03";grep "08/01-03" smb_wild.txt | wc -l
echo -n "08/01-04";grep "08/01-04" smb_wild.txt | wc -l
echo -n "08/01-05";grep "08/01-05" smb_wild.txt | wc -l
echo -n "08/01-06";grep "08/01-06" smb_wild.txt | wc -l
echo -n "08/01-07";grep "08/01-07" smb_wild.txt | wc -l
echo -n "08/01-08";grep "08/01-08" smb_wild.txt | wc -l
echo -n "08/01-09";grep "08/01-09" smb_wild.txt | wc -l
echo -n "08/01-10";grep "08/01-10" smb_wild.txt | wc -l
echo -n "08/01-11";grep "08/01-11" smb_wild.txt | wc -l
echo -n "08/01-12";grep "08/01-12" smb_wild.txt | wc -l
echo -n "08/01-13";grep "08/01-13" smb_wild.txt | wc -l
echo -n "08/01-14";grep "08/01-14" smb_wild.txt | wc -l
echo -n "08/01-15";grep "08/01-15" smb_wild.txt | wc -l
echo -n "08/01-16";grep "08/01-16" smb_wild.txt | wc -l
echo -n "08/01-17";grep "08/01-17" smb_wild.txt | wc -l
echo -n "08/01-18";grep "08/01-18" smb_wild.txt | wc -l
echo -n "08/01-19";grep "08/01-19" smb_wild.txt | wc -l
echo -n "08/01-20";grep "08/01-20" smb_wild.txt | wc -l
echo -n "08/01-21";grep "08/01-21" smb_wild.txt | wc -l
echo -n "08/01-22";grep "08/01-22" smb_wild.txt | wc -l
echo -n "08/01-23";grep "08/01-23" smb_wild.txt | wc -l
echo -n "08/02-00";grep "08/02-00" smb_wild.txt | wc -l
echo -n "08/02-01";grep "08/02-01" smb_wild.txt | wc -l
echo -n "08/02-02";grep "08/02-02" smb_wild.txt | wc -l
echo -n "08/02-03";grep "08/02-03" smb_wild.txt | wc -l
echo -n "08/02-04";grep "08/02-04" smb_wild.txt | wc -l
echo -n "08/02-05";grep "08/02-05" smb_wild.txt | wc -l
echo -n "08/02-06";grep "08/02-06" smb_wild.txt | wc -l
echo -n "08/02-07";grep "08/02-07" smb_wild.txt | wc -l
echo -n "08/02-08";grep "08/02-08" smb_wild.txt | wc -l
echo -n "08/02-09";grep "08/02-09" smb_wild.txt | wc -l
echo -n "08/02-10";grep "08/02-10" smb_wild.txt | wc -l
echo -n "08/02-11";grep "08/02-11" smb_wild.txt | wc -l
echo -n "08/02-12";grep "08/02-12" smb_wild.txt | wc -l
echo -n "08/02-13";grep "08/02-13" smb_wild.txt | wc -l
echo -n "08/02-14";grep "08/02-14" smb_wild.txt | wc -l
echo -n "08/02-15";grep "08/02-15" smb_wild.txt | wc -l
echo -n "08/02-16";grep "08/02-16" smb_wild.txt | wc -l
echo -n "08/02-17";grep "08/02-17" smb_wild.txt | wc -l
echo -n "08/02-18";grep "08/02-18" smb_wild.txt | wc -l
echo -n "08/02-19";grep "08/02-19" smb_wild.txt | wc -l
echo -n "08/02-20";grep "08/02-20" smb_wild.txt | wc -l
echo -n "08/02-21";grep "08/02-21" smb_wild.txt | wc -l
echo -n "08/02-22";grep "08/02-22" smb_wild.txt | wc -l
echo -n "08/02-23";grep "08/02-23" smb_wild.txt | wc -l
echo -n "08/03-00";grep "08/03-00" smb_wild.txt | wc -l
echo -n "08/03-01";grep "08/03-01" smb_wild.txt | wc -l
echo -n "08/03-02";grep "08/03-02" smb_wild.txt | wc -l
echo -n "08/03-03";grep "08/03-03" smb_wild.txt | wc -l
echo -n "08/03-04";grep "08/03-04" smb_wild.txt | wc -l
echo -n "08/03-05";grep "08/03-05" smb_wild.txt | wc -l
echo -n "08/03-06";grep "08/03-06" smb_wild.txt | wc -l
echo -n "08/03-07";grep "08/03-07" smb_wild.txt | wc -l
echo -n "08/03-08";grep "08/03-08" smb_wild.txt | wc -l
echo -n "08/03-09";grep "08/03-09" smb_wild.txt | wc -l
echo -n "08/03-10";grep "08/03-10" smb_wild.txt | wc -l
echo -n "08/03-11";grep "08/03-11" smb_wild.txt | wc -l
echo -n "08/03-12";grep "08/03-12" smb_wild.txt | wc -l
echo -n "08/03-13";grep "08/03-13" smb_wild.txt | wc -l
echo -n "08/03-14";grep "08/03-14" smb_wild.txt | wc -l
echo -n "08/03-15";grep "08/03-15" smb_wild.txt | wc -l
echo -n "08/03-16";grep "08/03-16" smb_wild.txt | wc -l
echo -n "08/03-17";grep "08/03-17" smb_wild.txt | wc -l
echo -n "08/03-18";grep "08/03-18" smb_wild.txt | wc -l
echo -n "08/03-19";grep "08/03-19" smb_wild.txt | wc -l
echo -n "08/03-20";grep "08/03-20" smb_wild.txt | wc -l
echo -n "08/03-21";grep "08/03-21" smb_wild.txt | wc -l
echo -n "08/03-22";grep "08/03-22" smb_wild.txt | wc -l
echo -n "08/03-23";grep "08/03-23" smb_wild.txt | wc -l
echo -n "08/04-00";grep "08/04-00" smb_wild.txt | wc -l
echo -n "08/04-01";grep "08/04-01" smb_wild.txt | wc -l
echo -n "08/04-02";grep "08/04-02" smb_wild.txt | wc -l
echo -n "08/04-03";grep "08/04-03" smb_wild.txt | wc -l
echo -n "08/04-04";grep "08/04-04" smb_wild.txt | wc -l
echo -n "08/04-05";grep "08/04-05" smb_wild.txt | wc -l
echo -n "08/04-06";grep "08/04-06" smb_wild.txt | wc -l
echo -n "08/04-07";grep "08/04-07" smb_wild.txt | wc -l
echo -n "08/04-08";grep "08/04-08" smb_wild.txt | wc -l
echo -n "08/04-09";grep "08/04-09" smb_wild.txt | wc -l
echo -n "08/04-10";grep "08/04-10" smb_wild.txt | wc -l
echo -n "08/04-11";grep "08/04-11" smb_wild.txt | wc -l
echo -n "08/04-12";grep "08/04-12" smb_wild.txt | wc -l
echo -n "08/04-13";grep "08/04-13" smb_wild.txt | wc -l
echo -n "08/04-14";grep "08/04-14" smb_wild.txt | wc -l
echo -n "08/04-15";grep "08/04-15" smb_wild.txt | wc -l
echo -n "08/04-16";grep "08/04-16" smb_wild.txt | wc -l
echo -n "08/04-17";grep "08/04-17" smb_wild.txt | wc -l
echo -n "08/04-18";grep "08/04-18" smb_wild.txt | wc -l
echo -n "08/04-19";grep "08/04-19" smb_wild.txt | wc -l
echo -n "08/04-20";grep "08/04-20" smb_wild.txt | wc -l
echo -n "08/04-21";grep "08/04-21" smb_wild.txt | wc -l
echo -n "08/04-22";grep "08/04-22" smb_wild.txt | wc -l
echo -n "08/04-23";grep "08/04-23" smb_wild.txt | wc -l
echo -n "08/05-00";grep "08/05-00" smb_wild.txt | wc -l
echo -n "08/05-01";grep "08/05-01" smb_wild.txt | wc -l
echo -n "08/05-02";grep "08/05-02" smb_wild.txt | wc -l
echo -n "08/05-03";grep "08/05-03" smb_wild.txt | wc -l
echo -n "08/05-04";grep "08/05-04" smb_wild.txt | wc -l
echo -n "08/05-05";grep "08/05-05" smb_wild.txt | wc -l
echo -n "08/05-06";grep "08/05-06" smb_wild.txt | wc -l
echo -n "08/05-07";grep "08/05-07" smb_wild.txt | wc -l
echo -n "08/05-08";grep "08/05-08" smb_wild.txt | wc -l
echo -n "08/05-09";grep "08/05-09" smb_wild.txt | wc -l
echo -n "08/05-10";grep "08/05-10" smb_wild.txt | wc -l
echo -n "08/05-11";grep "08/05-11" smb_wild.txt | wc -l
echo -n "08/05-12";grep "08/05-12" smb_wild.txt | wc -l
echo -n "08/05-13";grep "08/05-13" smb_wild.txt | wc -l
echo -n "08/05-14";grep "08/05-14" smb_wild.txt | wc -l
echo -n "08/05-15";grep "08/05-15" smb_wild.txt | wc -l
echo -n "08/05-16";grep "08/05-16" smb_wild.txt | wc -l
echo -n "08/05-17";grep "08/05-17" smb_wild.txt | wc -l
echo -n "08/05-18";grep "08/05-18" smb_wild.txt | wc -l
echo -n "08/05-19";grep "08/05-19" smb_wild.txt | wc -l
echo -n "08/05-20";grep "08/05-20" smb_wild.txt | wc -l
echo -n "08/05-21";grep "08/05-21" smb_wild.txt | wc -l
echo -n "08/05-22";grep "08/05-22" smb_wild.txt | wc -l
echo -n "08/05-23";grep "08/05-23" smb_wild.txt | wc -l
```

The output of the script was then again import into Excel to build the distribution table.

A modified script from Royans Tharakan (http://www.giac.org/practical/Royans_Tharakan_GCIA.doc) was also used to extract other details from concatenated log files.

```
#!/bin/bash
# Scripts from Royans_Tharakan_GCIA.doc

echo "Top Alerts"
#cat alert_all  | perl -e 'while(<STDIN>) {@a=split(/\[\*\*\]/,$_);print "$a[1]\n";}'  | sor
t -nr | uniq -c > top_alerts.txt

echo "Top Source IP Addresses"
#cat alert_all | perl -e 'while(<STDIN>) {@a=split(/\[\*\*\]/,$_);@b=split(/ /,$a[2]);@c=spl
it(/:/,$b[1]);print "$c[0]\n";}'  | sort -nr | uniq -c | sort -nr > top_source_ip.txt

echo "Top Source ports"
#cat alert_all | perl -e 'while(<STDIN>) {@a=split(/\[\*\*\]/,$_);@b=split(/ /,$a[2]);@c=spl
it(/:/,$b[1]);print "$c[1]\n";}'  | sort -nr | uniq -c | sort -nr > top_source_port.txt

echo "Top Destination IPs"
#cat alert_all | perl -e 'while(<STDIN>) {@a=split(/\[\*\*\]/,$_);@b=split(/ /,$a[2]);@c=spl
it(/:/,$b[3]);print "$c[0]\n";}' | sort -nr | uniq -c | sort -nr > top_dest_ip.txt

echo "Top Destination ports"
#cat alert_all | perl -e 'while(<STDIN>) {@a=split(/\[\*\*\]/,$_);@b=split(/ /,$a[2]);@c=spl
it(/:/,$b[3]);print "$c[1]\n";}'  | sort -nr | uniq -c | sort -nr > top_dest_port.txt

echo "Top Portscaning hosts \- probably compromised systems"
cat scans.logs | grep -v UDP | awk '{print $4}' | cut -d":" -f1| sort -nr | uniq -c | sort -
nr > top_scan_host.txt

echo "Top OOS (Out of Spec) Source Ips"
cat oos.logs | grep "^08" | awk '{print $2}' | cut -d":" -f1 | sort -nr | uniq -c | sort -n
> top_oos_source_ip.txt

echo "Top OOS (Out of Spec) Source Ports"
cat oos.logs | grep "^08" | awk '{print $2}' | cut -d":" -f2 | sort -nr | uniq -c | sort -n
> top_oos_source_port.txt

echo "Top OOS (Out of Spec) Destination Ips"
cat oos.logs | grep "^08" | awk '{print $4}' | cut -d":" -f1 | sort -nr | uniq -c | sort -n
> top_oos_dest_ip.txt

echo "Top OOS (Out of Spec) Destination Ports"
cat oos.logs | grep "^08" | awk '{print $4}' | cut -d":" -f2 | sort -nr | uniq -c | sort -n
> top_oos_dest_port.txt
```

Logs of individual IP addresses were extracted using standard grep and awk commands.

To draw the link graph, I used a tools called "SmartDraw".

To retrieve registrant information of external IP addresses, I used a tools called 'SmartWhois".

References

Caswell and Anuzis. “BACKDOOR subseven 22.” Snort Signatures Database. Jan 30, 2002. URL: http://www.snort.org/snort-db/sid.html?id=103. (18 Oct 2002)

Cisco Systems, Inc. “How to Protect Your Network Against the Nimda Virus.” Feb 18, 2002. URL: http://www.cisco.com/warp/public/63/nimda.shtml . (18 Oct 2002)

Internet Assigned Numbers Authority [IANA]. “Port Numbers.” May 2, 2002. URL: http://www.iana.org/assignments/port-numbers . (18 Oct 2002)

Sourcefire, Inc. “Snort Ports Database.” URL: http://www.snort.org/ports.html . (18 Oct 2002)

Snort.org. Snort Users Manual. URL: http://www.snort.org/docs/writing_rules/ (18 Oct 2002)

Vision, Max. “Re: [snort] ‘SMB Name Wildcard.’” Archives.Neohapsis.Com. Jan 17, 2000. URL: http://archives.neohapsis.com/archives/snort/2000-01/0220.html . (18 Oct 2002)

Tod A. Beardsley. “Intrusion Detection and Analysis: Theory, Techniques, and Tools” GIAC Certified Intrusion Analysts (GCIA). URL: http://www.giac.org/practical/Tod_Beardsley_GCIA.doc. (18 Oct 2002)

Brannan, Andrew. “Unicode Vulnerability – How & Why?” 7 August 2001. URL: http://rr.sans.org/threats/unicode.php . (18 Oct 2002)

Mark Wilson. “Intrusion Detection In Depth” GIAC Certified Intrusion Analysts (GCIA). URL: http://www.giac.org/practical/mark_wilson_gcia.zip (18 Oct 2002)

CERT. CERT® Advisory CA-2001-26 Nimda Worm. URL: http://www.cert.org/advisories/CA-2001-26.html. (18 Oct 2002)

CERT. CERT Advisory CA-2001-13 Buffer Overflow In IIS Indexing Service DLL. URL: http://www.cert.org/advisories/CA-2001-13.html. (18 Oct 2002)

Alexander, Bryce. “Intrusion Detection FAQ Port 137 Scan”. May 10, 2000. URL: http://www.sans.org/newlook/resources/IDFAQ/port_137.htm. (18 Oct 2002)

Steven Drew. “Intrusion Detection In Depth” GIAC Certified Intrusion Analysts (GCIA). URL: http://www.giac.org/practical/Steven_Drew_GCIA.doc (18 Oct 2002)

Royans Tharakan . “Intrusion Detection In Depth” GIAC Certified Intrusion Analysts (GCIA). URL: http://www.giac.org/practical/Royans_Tharakan_GCIA.doc (18 Oct 2002)

News Riders. "Network Intrusion Detection. An Analyst's Handbook, Second Edition" by Stephen Northcutt and Judy Novak. ISBN 0-7357-1008-2.

News Riders. "Intrusion Signatures and Analysis" by Stephen Northcutt, MarkCooper, Matt Fearnow and Karen Frederick. ISBN-07357-1063-5.