# Analyzing Cyber Threats at a Public WIFI Hotspot

Naif Alqramin
Lewis University
MSIS 2011

## Table of Contents

## Introduction

This project seeks to analyze data captured from a public WIFI hotspot and to interpret each alert using a suite of smart tools to help determine the nature of the alerts. The tools used include Wireshark, Snort, Netwitness, Whois Command, Side Jacking, and others. Although, each one of these tools has its own strengths and weaknesses, combining them together is a great idea to solve the puzzle.

I used a local Starbucks as my open Wifi hotspot. I collected a174 MB capture in one hour and 50 minutes. Snort registered 566 alerts; 27 of them were unique alerts. By looking at the traffic profile by protocol, I found out that TCP takes 46%, UDP with 0%, ICMP takes the highest percentage with 53%, and finally port scan takes only 1%. I was surprised by the result that a place with open connection and unidentified users could cause this large of a number of alerts in such a short amount of time. However, I realized that I need to dig deep inside these alerts, in order to look for a number of facts such as what is the root cause of the problem? Was it generated by an automatic tool or manually? Is it serious in nature, or just a false positive? Finally, I would suggest several procedures and polices which will help businesses running open WIFI hotspots to protect their customers and valuable assets.

## Executive Summary

This paper will analyze all of the different kinds of threats that were recorded during the listening session, Snort registered 566 alerts; 27 of them were unique cyber security threats. I will focus on these 27 unique alerts and discuss the following facts; the type of the threat, along with their percentage representation, description, attack scenario, the seriousness of the alert, root cause of the incident, if possible, who initiated that threat and its recipients, and finally the recommended action for a public hotspot wifi administrators. Regarding the root cause of each threat point, I identify whether a threat is automatically generated by a tool (software) or manually generated by human action. Further, timing techniques were used to determine whether the threat was generated automatically or manually. If there were only one or two connection attempts coming from a particular IP address source, then it would be probably manually generated by a human. However, if the connection is coming rapidly and at regular intervals, for example every single second, it means the root cause of the attack is generated automatically by a hacking or scanning tool (software).

Figure 1 summarizes the essential characteristics of the study.

| | |
|---|---|
| **Tools** | **Wireshark, Snort, Netwitness, Whois Command Tools** |
| **Capture Size** | **174 MB** |
| **Time** | **Crowded Hours Between 5:00 7:00 PM Weekend Day** |
| **Duration** | **Duration of the Listening Session is 2 hours** |
| **Place** | **Capture took place at a Startbucks Coffee Company Branch** |

Figure 1: Details of this study.

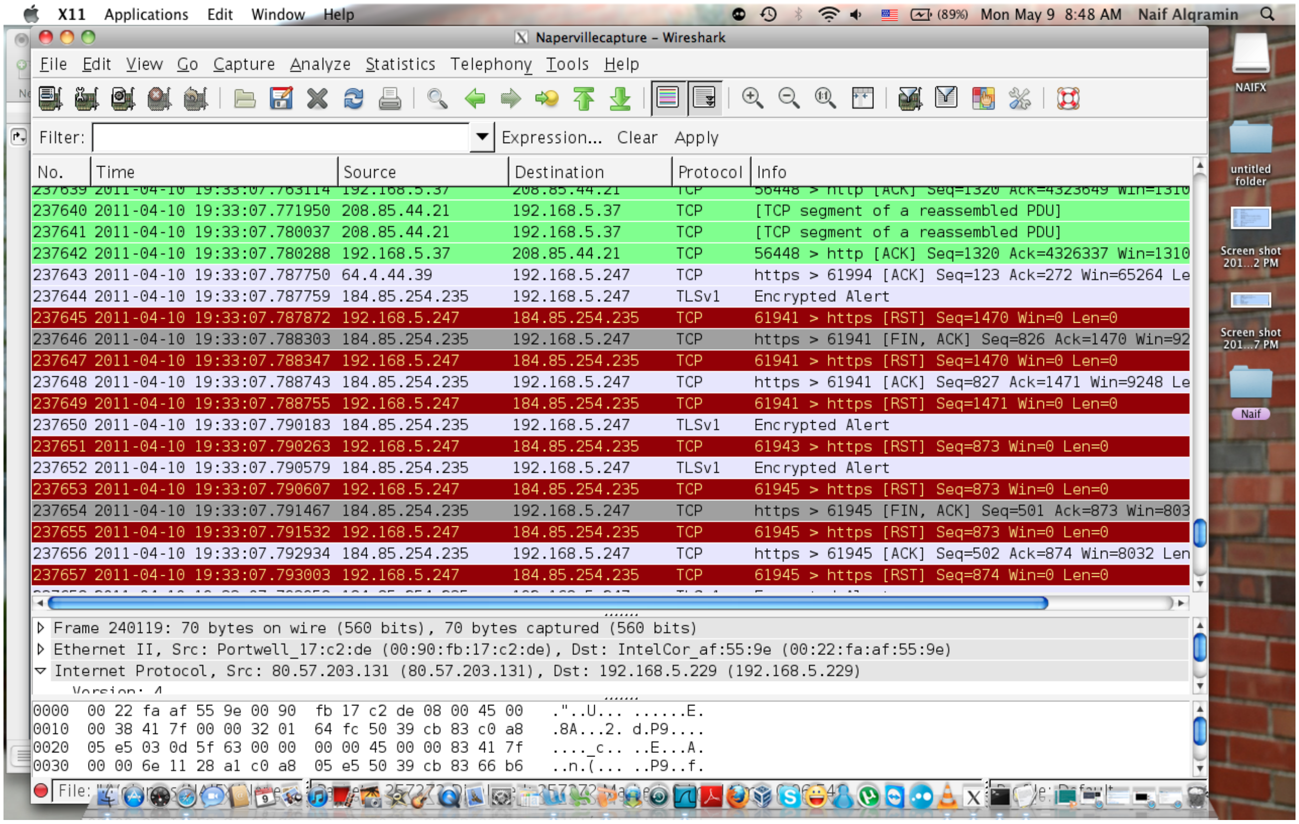
# Chapter 2: Tool Descriptions

This chapter introduces the various tools that were used to collect and analyze the wireless data. A good understanding of what these tools do and how they are used is essential to appreciating the meaning of the collected data.

## Wireshark Tool

I will start with Wireshark tool. Wireshark is an open-source tool for profiling network traffic and analyzing packets. Wireshark, formerly known as Ethereal, can be used to examine the details of traffic at a variety of levels ranging from connection-level information to the bits that make up a single packet. In addition, packet capture can provide a network administrator with information about individual packets such as transmit time, source, destination, protocol type and Header data. This information can be useful for evaluating security events and troubleshooting network security device issues. Wireshark will typically display information in three panels. The top panel lists frames individually with key data on a single line. Any single frame selected in the top pane is further explained in the tool's middle panel. In this section of the display, Wireshark shows packet details, illustrating how various aspects of the frame can be understood as belonging to the Data Link Layer, Network Layer, Transport Layer or Application Layer. See figure 2 for an example.

Since Wireshark’s wireless analysis features have grown to be an especially powerful tool for capturing, and analyzing wireless networks, therefore, I planned to use it to be my listening session tool for my project. With Wireshark’s view filters and resilient protocol dissector features, an administrator can sift through large quantities of wireless traffic to identify a specific condition or field value being looked for, or to exclude undesirable traffic until are only a handful of traffic remains to be assessed. [1,2]


X11 Applications Edit Window Help
(89%) Mon May 9 8:48 AM Naif Alqramin
Napervillecapture - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
NAIFX
untitled folder
Screen shot 201...2 PM
Screen shot 201...7 PM
Naif

```
No.    Time                       Source          Destination     Protocol  Info
237639 2011-04-10 19:33:07.763114 192.168.5.37    208.85.44.21    TCP       56448 > http [ACK] Seq=1320 Ack=4323649 Win=1310
237640 2011-04-10 19:33:07.771950 208.85.44.21    192.168.5.37    TCP       [TCP segment of a reassembled PDU]
237641 2011-04-10 19:33:07.780037 208.85.44.21    192.168.5.37    TCP       [TCP segment of a reassembled PDU]
237642 2011-04-10 19:33:07.780288 192.168.5.37    208.85.44.21    TCP       56448 > http [ACK] Seq=1320 Ack=4326337 Win=1310
237643 2011-04-10 19:33:07.787750 64.4.44.39      192.168.5.247   TCP       https > 61994 [ACK] Seq=123 Ack=272 Win=65264 Le
237644 2011-04-10 19:33:07.787759 184.85.254.235  192.168.5.247   TLSv1     Encrypted Alert
237645 2011-04-10 19:33:07.787872 192.168.5.247   184.85.254.235  TCP       61941 > https [RST] Seq=1470 Win=0 Len=0
237646 2011-04-10 19:33:07.788303 184.85.254.235  192.168.5.247   TCP       https > 61941 [FIN, ACK] Seq=826 Ack=1470 Win=92
237647 2011-04-10 19:33:07.788347 192.168.5.247   184.85.254.235  TCP       61941 > https [RST] Seq=1470 Win=0 Len=0
237648 2011-04-10 19:33:07.788743 184.85.254.235  192.168.5.247   TCP       https > 61941 [ACK] Seq=827 Ack=1471 Win=9248 Le
237649 2011-04-10 19:33:07.788755 192.168.5.247   184.85.254.235  TCP       61941 > https [RST] Seq=1471 Win=0 Len=0
237650 2011-04-10 19:33:07.790183 184.85.254.235  192.168.5.247   TLSv1     Encrypted Alert
237651 2011-04-10 19:33:07.790263 192.168.5.247   184.85.254.235  TCP       61943 > https [RST] Seq=873 Win=0 Len=0
237652 2011-04-10 19:33:07.790579 184.85.254.235  192.168.5.247   TLSv1     Encrypted Alert
237653 2011-04-10 19:33:07.790607 192.168.5.247   184.85.254.235  TCP       61945 > https [RST] Seq=873 Win=0 Len=0
237654 2011-04-10 19:33:07.791467 184.85.254.235  192.168.5.247   TCP       https > 61945 [FIN, ACK] Seq=501 Ack=873 Win=803
237655 2011-04-10 19:33:07.791532 192.168.5.247   184.85.254.235  TCP       61945 > https [RST] Seq=873 Win=0 Len=0
237656 2011-04-10 19:33:07.792934 184.85.254.235  192.168.5.247   TCP       https > 61945 [ACK] Seq=502 Ack=874 Win=8032 Len
237657 2011-04-10 19:33:07.793003 192.168.5.247   184.85.254.235  TCP       61945 > https [RST] Seq=874 Win=0 Len=0
```

```
▷ Frame 240119: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
▷ Ethernet II, Src: Portwell_17:c2:de (00:90:fb:17:c2:de), Dst: IntelCor_af:55:9e (00:22:fa:af:55:9e)
▽ Internet Protocol, Src: 80.57.203.131 (80.57.203.131), Dst: 192.168.5.229 (192.168.5.229)
    Version: 4
```

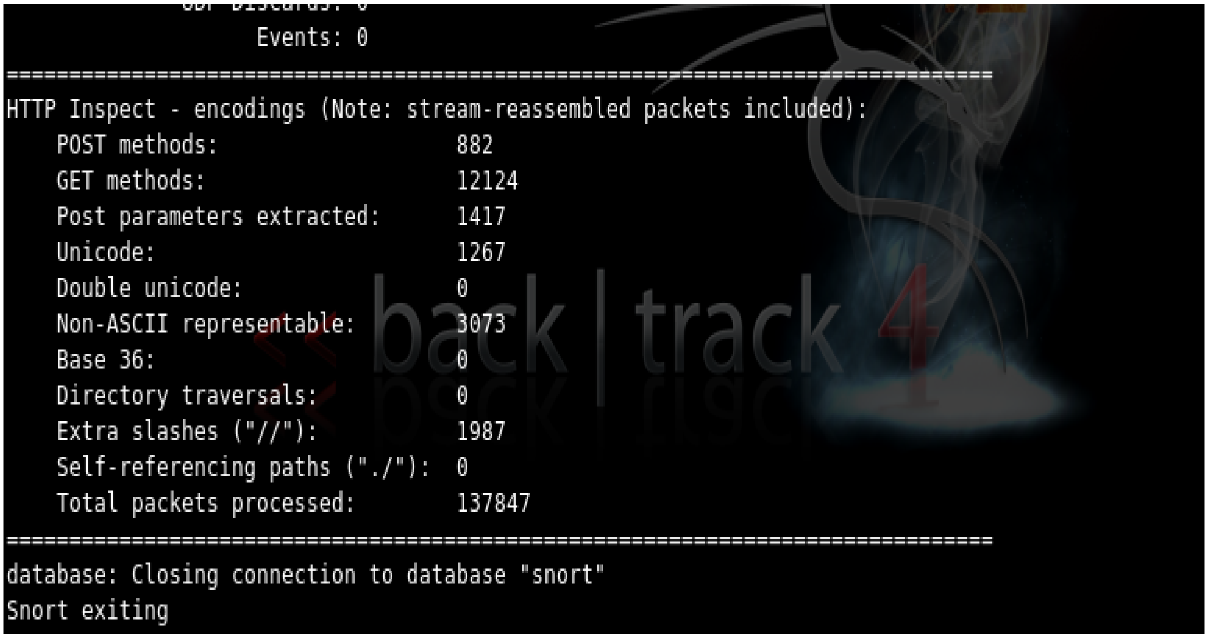
```
0000  00 22 fa af 55 9e 00 90  fb 17 c2 de 08 00 45 00   ."..U... ......E.
0010  00 38 41 7f 00 00 32 01  64 fc 50 39 cb 83 c0 a8   .8A...2. d.P9....
0020  05 e5 03 0d 5f 63 00 00  00 00 45 00 00 83 41 7f   ...._c.. ..E...A.
0030  00 00 6e 11 28 a1 c0 a8  05 e5 50 39 cb 83 66 b6   ..n.(... ..P9..f.
```

Figure 2: Wireshark

## Snort IDSP Tool

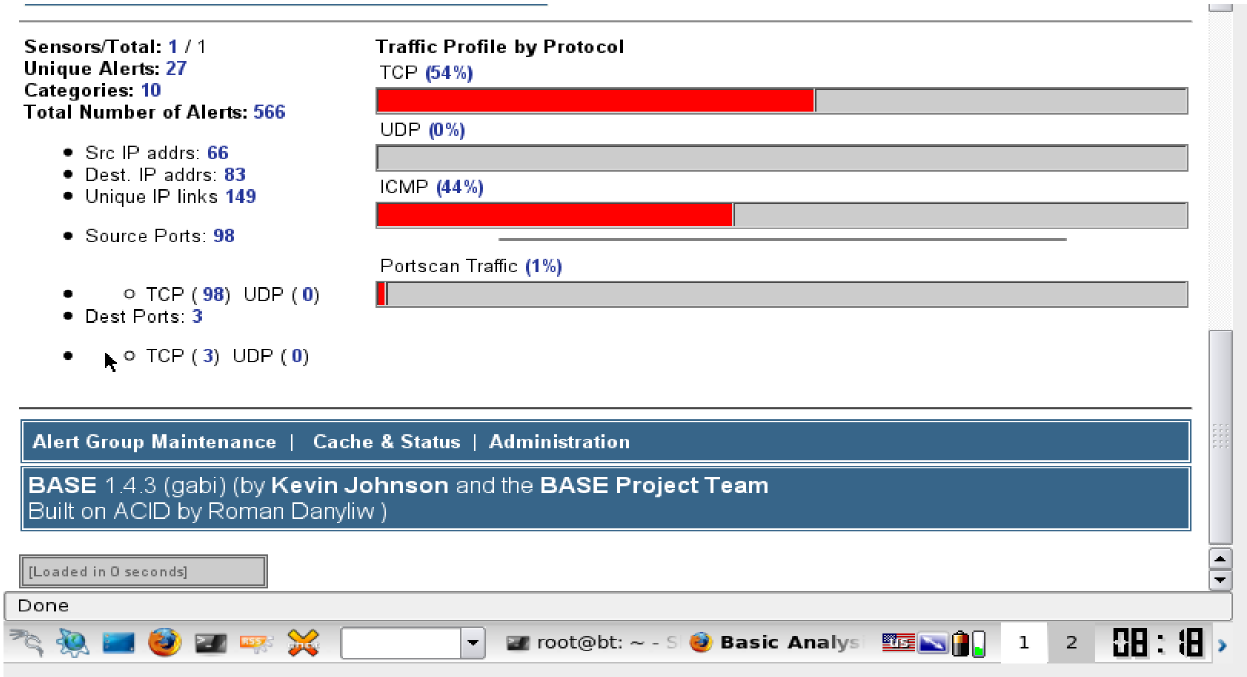
I intentionally used Snort tool as my Intrusion Detection & Prevention System (IDPS) for powerfully analyzing data packages. IDPS is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which includes violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices. Snort performs intrusion detection role and attempts to stop detected possible undesirable incidents. Snort is primarily focused on identifying possible incidents, logging information about them, attempting to stop them, and reporting them to security administrators or Network top management. See figure 3 and 4 for an example.

In addition, organizations use Snort for other purposes, such as identifying problems with security policies, documenting existing threats, and deterring individuals from violating security policies. It has become a necessary addition to the security infrastructure of nearly every organization. The IDS preprocessor in Snort typically records information related to observed events, notifies security administrators of important observed events, and produces reports. Many IDS can also respond to a detected threat by attempting to prevent it from succeeding. They use several response techniques, which involve the IDPS stopping the attack itself, changing the security environment, reconfiguring the firewall, or changing the attack's content. [3,4]



```
              UDP Discards: 0
                    Events: 0
===============================================================================
HTTP Inspect - encodings (Note: stream-reassembled packets included):
    POST methods:                   882
    GET methods:                    12124
    Post parameters extracted:      1417
    Unicode:                        1267
    Double unicode:                 0
    Non-ASCII representable:        3073
    Base 36:                        0
    Directory traversals:           0
    Extra slashes ("//"):           1987
    Self-referencing paths ("./"):  0
    Total packets processed:        137847
===============================================================================
database: Closing connection to database "snort"
Snort exiting
```

Figure 3: Snort Command-line



**Sensors/Total:** 1 / 1
**Unique Alerts:** 27
**Categories:** 10
**Total Number of Alerts:** 566

- Src IP addrs: 66
- Dest. IP addrs: 83
- Unique IP links 149
- Source Ports: 98
- ○ TCP ( 98) UDP ( 0)
- Dest Ports: 3
- ○ TCP ( 3) UDP ( 0)

**Traffic Profile by Protocol**

TCP (54%)

UDP (0%)

ICMP (44%)

Portscan Traffic (1%)

**Alert Group Maintenance** | **Cache & Status** | **Administration**

**BASE** 1.4.3 (gabi) (by **Kevin Johnson** and the **BASE Project Team**
Built on ACID by Roman Danyliw )

[Loaded in 0 seconds]

Done

Figure 4: Snort, and ACID Database

## Netwitness Investigator Tool

Netwitness Investigator is interactive threat analysis to solve a wide range of challenging information security problems including: insider threats, zero-day exploits and targeted malware, advanced persistent threats, fraud, espionage, data leakage, and continuous monitoring of security controls. Netwitness Investigator provides security operations staff, auditors, and fraud and forensics investigators the power to perform unprecedented free-form contextual analysis of raw network data.

However, both beginner and expert users can use this software to powerfully grab huge amounts of network traffic easily to dive deeply into the context and content of network sessions in real-time, shortening threat analysis into minutes instead of days. In addition to the rich data the examiner receives from the Netwitness infrastructure, the examiner can locally capture live traffic and process packet files from virtually any accessible network collection device for quick and easy analysis. See figure 5 for an example. And by integrating Netwitness Investigator Enterprise with Netwitness Live, you also have real-time fusion with multi-source threat intelligence. [5]

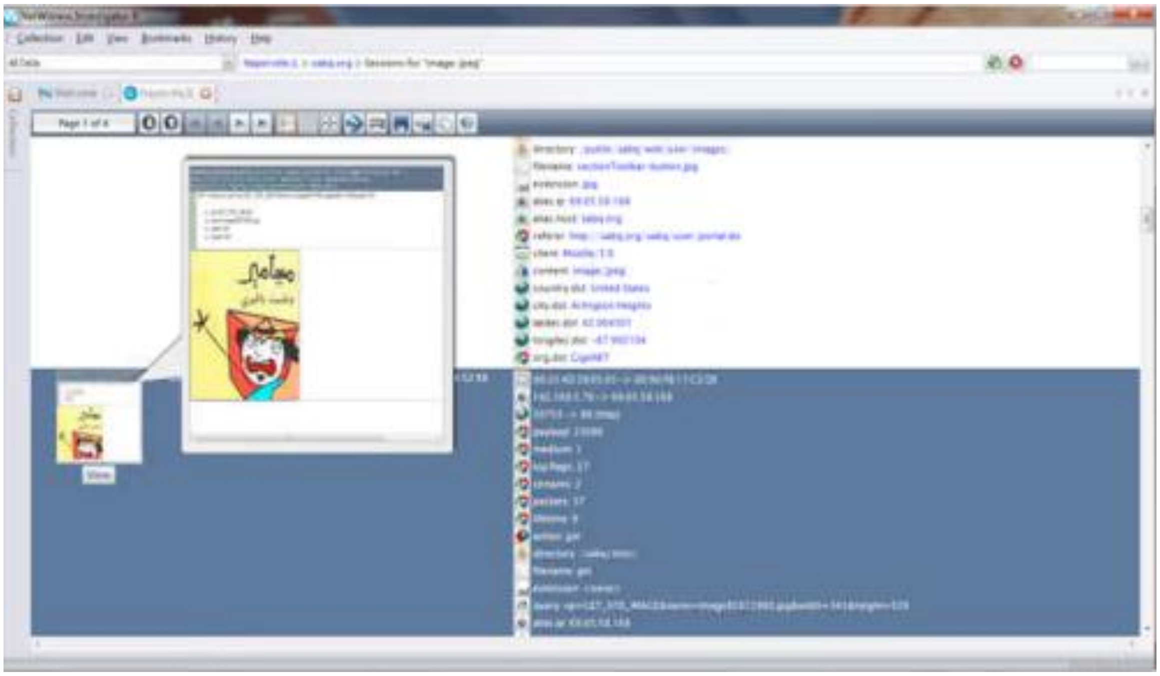
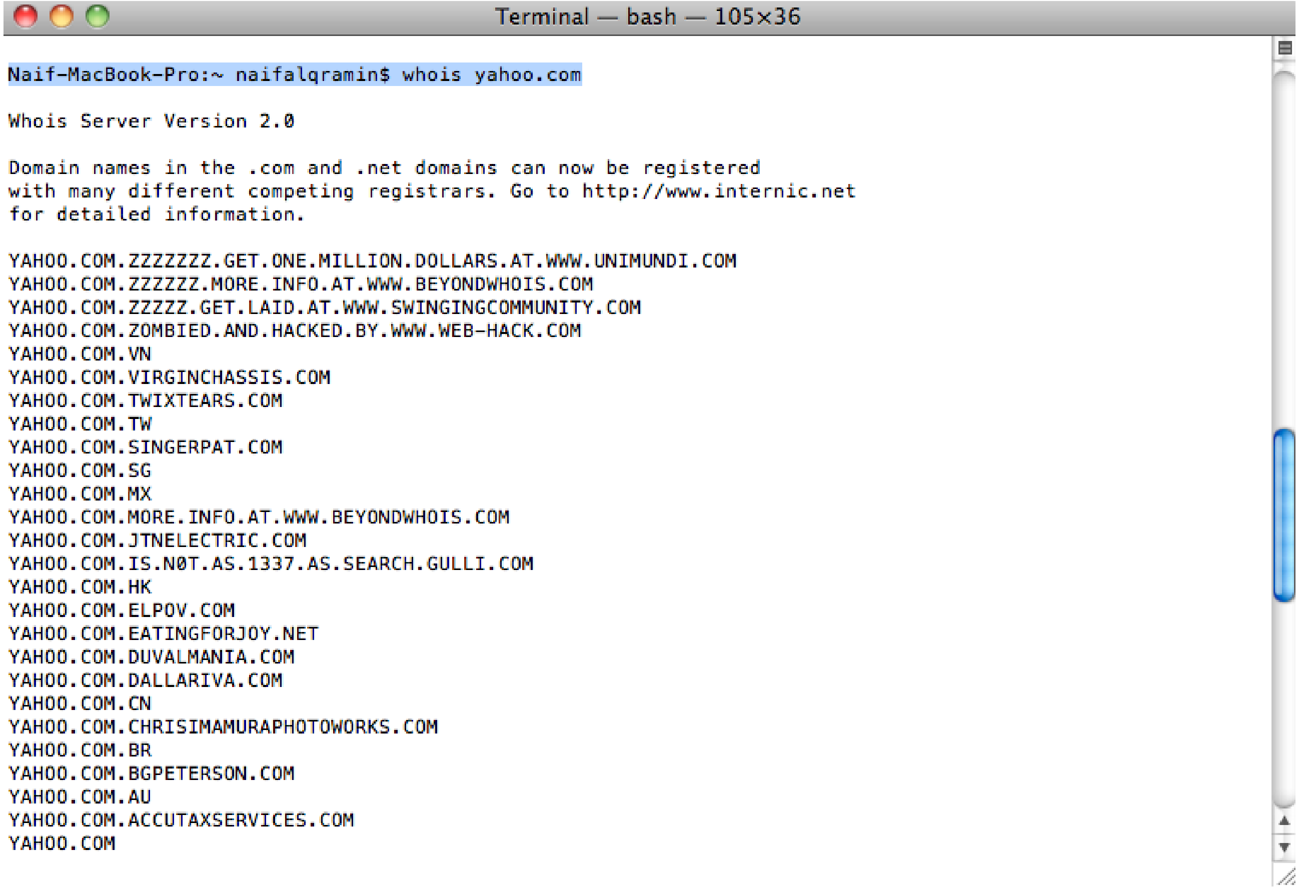


Figure 5: Netwintess Investigator

## Whois Command Line Tool

Whois Command Line is a simple command-line utility that allows administrators to easily get information about a registered domain. It automatically connects to the right WHOIS server, according to the top-level domain name, and retrieves the WHOIS record of the domain. It supports both generic domains and country code domains. See figure 6 for an example.



```
Naif-MacBook-Pro:~ naifalqramin$ whois yahoo.com

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

YAHOO.COM.ZZZZZZZ.GET.ONE.MILLION.DOLLARS.AT.WWW.UNIMUNDI.COM
YAHOO.COM.ZZZZZZ.MORE.INFO.AT.WWW.BEYONDWHOIS.COM
YAHOO.COM.ZZZZZ.GET.LAID.AT.WWW.SWINGINGCOMMUNITY.COM
YAHOO.COM.ZOMBIED.AND.HACKED.BY.WWW.WEB-HACK.COM
YAHOO.COM.VN
YAHOO.COM.VIRGINCHASSIS.COM
YAHOO.COM.TWIXTEARS.COM
YAHOO.COM.TW
YAHOO.COM.SINGERPAT.COM
YAHOO.COM.SG
YAHOO.COM.MX
YAHOO.COM.MORE.INFO.AT.WWW.BEYONDWHOIS.COM
YAHOO.COM.JTNELECTRIC.COM
YAHOO.COM.IS.N0T.AS.1337.AS.SEARCH.GULLI.COM
YAHOO.COM.HK
YAHOO.COM.ELPOV.COM
YAHOO.COM.EATINGFORJOY.NET
YAHOO.COM.DUVALMANIA.COM
YAHOO.COM.DALLARIVA.COM
YAHOO.COM.CN
YAHOO.COM.CHRISIMAMURAPHOTOWORKS.COM
YAHOO.COM.BR
YAHOO.COM.BGPETERSON.COM
YAHOO.COM.AU
YAHOO.COM.ACCUTAXSERVICES.COM
YAHOO.COM
```

Figure 6: Whois Command-Line Tool

# Chapter 3: Internet Control Message Protocol Cyber Threats

The section will focus on the Internet Control Message Protocol (ICMP) threats that are used in the Internet Architecture to perform the fault-isolation function, which is the group of actions that hosts and routers take to determine that there is a network failure. When an intermediate router detects a network problem while trying to forward an IP packet, it will usually send an ICMP error message to the source host, to raise awareness of the network problem. In the same way, there are a number of cases in which an end-system may generate an ICMP error message when it finds a problem while processing a datagram. These error messages are notified to the corresponding transport-protocol instance. [6]

In addition, this section will discuss relevant facts about each threat in order to get a better idea about the surrounding environment of each threat. The following will be discussed for each threat:

- The type of the threat
- Description of the threat
- The possible threat scenario
- The type of the alert
- The root cause of the incident
- Representative percentages
- If possible, who initiated that threat and its recipients
- The recommended action for administrator

### 1- ICMP Destination Unreachable Port Unreachable

This incident is generated when an Internet Control Message Protocol Port Unreachable message was detected. An ICMP Port Unreachable is not an attack, but may indicate that the source of the packet was the target of a scan or other malicious activity. An ICMP Port

Unreachable indicates that someone tried to connect to a port on a system that was not available or there is no service was running on that port.  This is analogous to RST packets in TCP. Since UDP does not have an equivalent, it relies upon ICMP Port Unreachable for this. This often indicates someone was scanning for UDP services. An attacker may use a port scanner to determine possible attack vectors as a prelude to a directed attack against a system. This kind of packet is common on networks, and may be generated by simple misconfigurations on either the source or destination, or service outage. [7]

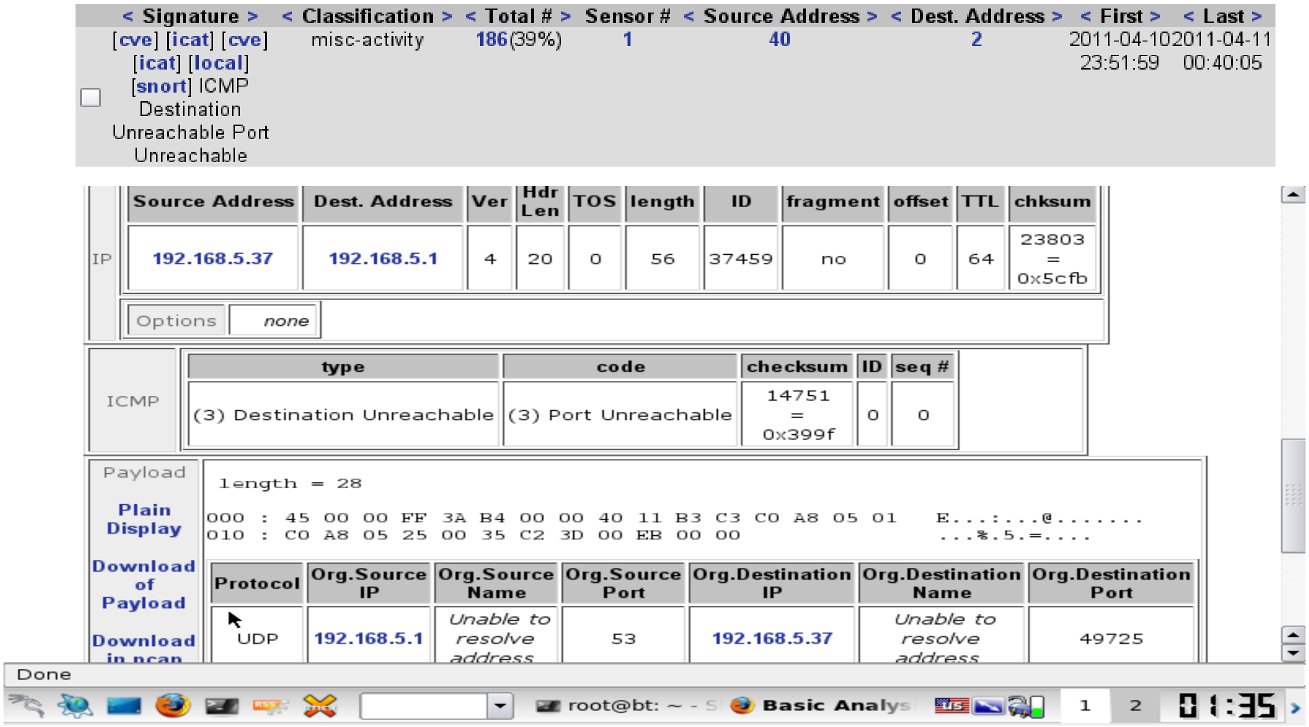
Network administrator should answer the following questions:

1- Are the host and the communications infrastructure working properly?

2- Is the ICMP Port Unreachable message originates from a host, not a router?

3- What the port is used for and why it wasn't available?

| ICMP Unreachable Port | Analysis |
|---|---|
| Kind of Alert | False Positive, but we need to block unused ports. Also, it is recommend to leave the alert ON with low priority tag. The priority tag assigns a severity level to rules, which are matching any kind of pattern. In this case administrator should leave it with low priority tag, in this case IP source keeps trying to reach unused port for a while, so we can do further serious action. For example; report the incident to top management to make appropriate decision, or file evidences to report that malicious behavior to authority. |
| Root cause &<br>Was it generated by an automatic tool or manually? | Generated manually, by connecting to a port on a system that was not available.<br>*** This can be determined by timing technique; if you see only one or two scans attempting to come from a particular source then it is probably manually generated. However, If the connection is coming rapidly and at regular intervals, for example every single second, it means the root cause of the problem is generated automatically by a hacking or scanning tool |
| Percentage | 39% |
| Whois Command | Both addresses "Source & Destination" are private |



| < Signature > | < Classification > | < Total # > | Sensor # | < Source Address > | < Dest. Address > | < First > | < Last > |
|---|---|---|---|---|---|---|---|
| □ [cve] [icat] [cve] [icat] [local] [snort] ICMP Destination Unreachable Port Unreachable | misc-activity | 186 (39%) | 1 | 40 | 2 | 2011-04-10 23:51:59 | 2011-04-11 00:40:05 |

IP

| Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|
| 192.168.5.37 | 192.168.5.1 | 4 | 20 | 0 | 56 | 37459 | no | 0 | 64 | 23803 = 0x5cfb |

| Options | none |
|---|---|

ICMP

| type | code | checksum | ID | seq # |
|---|---|---|---|---|
| (3) Destination Unreachable | (3) Port Unreachable | 14751 = 0x399f | 0 | 0 |

Payload

Plain Display

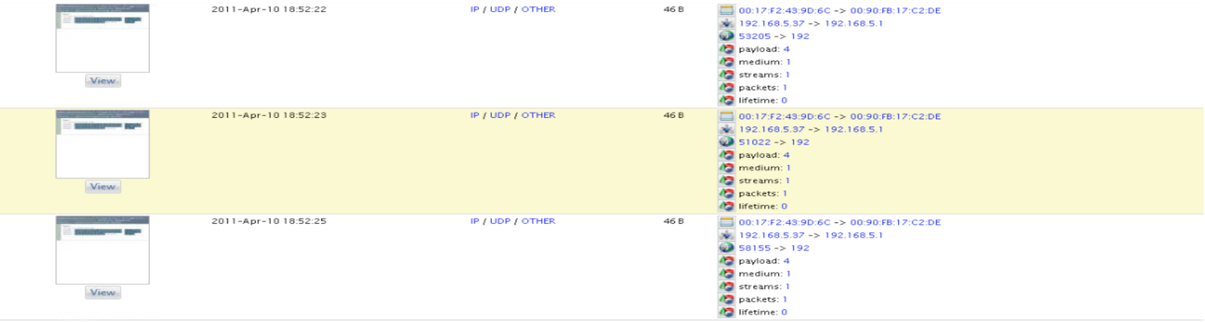
Download of Payload

Download in pcap

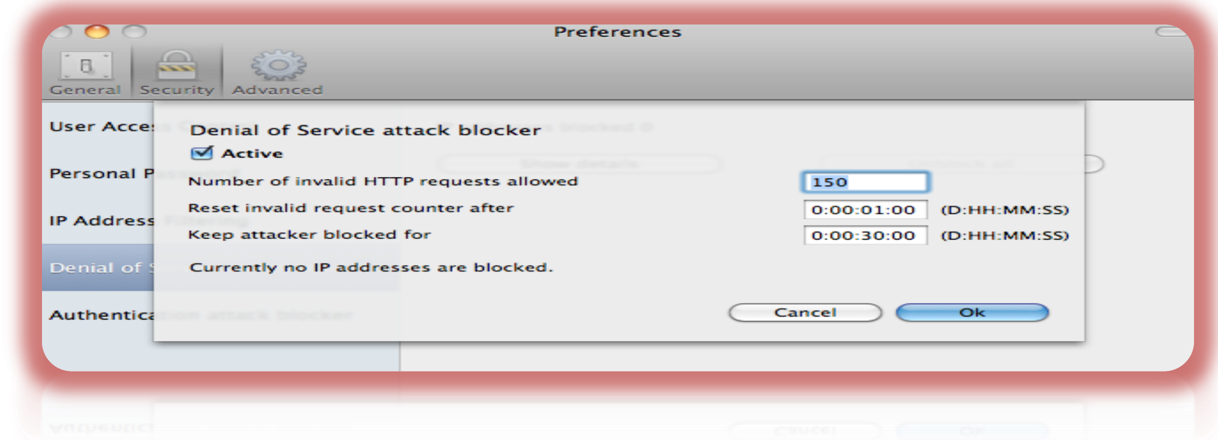
```
length = 28

000 : 45 00 00 FF 3A B4 00 00 40 11 B3 C3 C0 A8 05 01    E...:...@.......
010 : C0 A8 05 25 00 35 C2 3D 00 EB 00 00                 ...%.5.=....
```

| Protocol | Org.Source IP | Org.Source Name | Org.Source Port | Org.Destination IP | Org.Destination Name | Org.Destination Port |
|---|---|---|---|---|---|---|
| UDP | 192.168.5.1 | Unable to resolve address | 53 | 192.168.5.37 | Unable to resolve address | 49725 |

Done



| | | | | |
|---|---|---|---|---|
| View | 2011-Apr-10 18:52:22 | IP / UDP / OTHER | 46 B | 00:17:F2:43:9D:6C -> 00:90:FB:17:C2:DE<br>192.168.5.37 -> 192.168.5.1<br>53205 -> 192<br>payload: 4<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0 |
| View | 2011-Apr-10 18:52:23 | IP / UDP / OTHER | 46 B | 00:17:F2:43:9D:6C -> 00:90:FB:17:C2:DE<br>192.168.5.37 -> 192.168.5.1<br>51022 -> 192<br>payload: 4<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0 |
| View | 2011-Apr-10 18:52:25 | IP / UDP / OTHER | 46 B | 00:17:F2:43:9D:6C -> 00:90:FB:17:C2:DE<br>192.168.5.37 -> 192.168.5.1<br>58155 -> 192<br>payload: 4<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0 |



Denial of Service attack blocker

☑ Active

Number of invalid HTTP requests allowed: 150

Reset invalid request counter after: 0:00:01:00 (D:HH:MM:SS)

Keep attacker blocked for: 0:00:30:00 (D:HH:MM:SS)

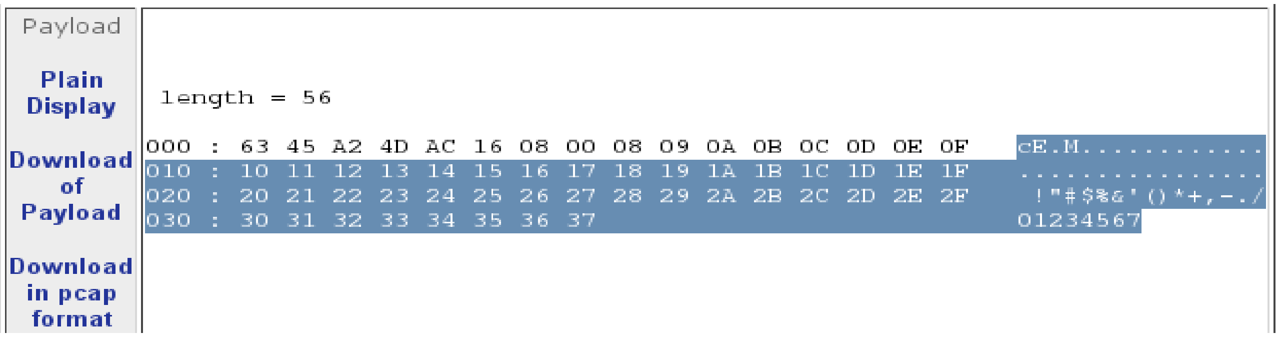
Currently no IP addresses are blocked.

Figure 7: ICMP Destination Unreachable Port attack screenshots

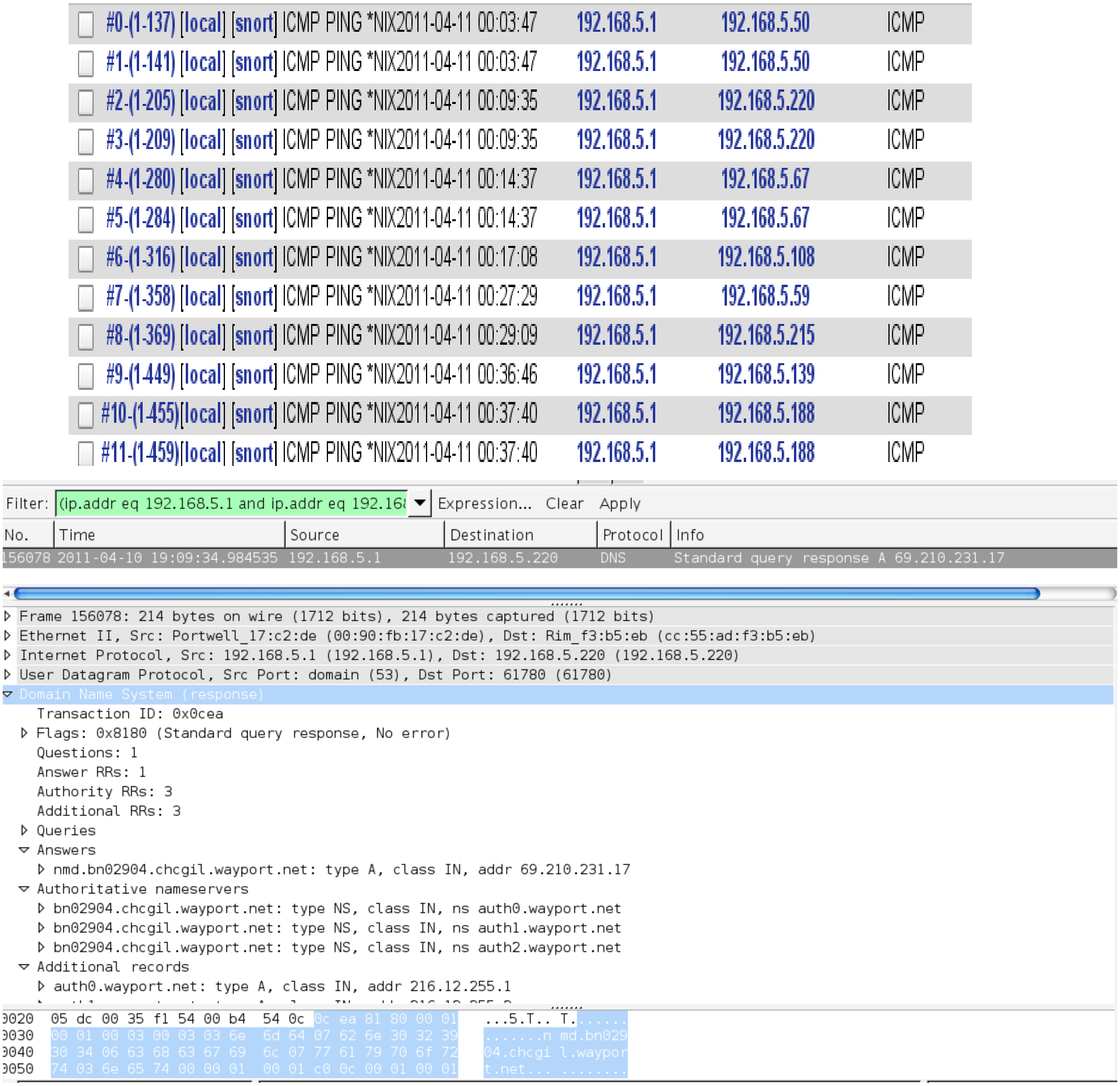
## 2- ICMP PING OR *NIX PING

This event indicates an ICMP echo request originating from the common utility known as “Ping”, often from a Unix platform operating system. This event is commonly used to measure the health and or availability of an IP protocol on a network connected device. The perverse use of the ICMP echo request could indicate an attacker, trying to map your network by seeing what hosts respond and what type of response is generated from these hosts to perform remote operating system identification. Ping is a standard networking utility that determines if a target host is up. Ping sends an ICMP echo request packet to an IP address. If a host is up at that address it will reply with an ICMP echo reply. The reply includes the data portion of the echo packet. The data included in the echo request varies across different operating system implementations. Ping can be used as a reconnaissance tool.  The impact of this event indicates an attempt to request the availability of a host, while in a paranoid mindset this could be viewed as a precursor to an upcoming attack. [8]

| ICMP Ping/Nix Ping | Analysis |
|---|---|
| Kind of Alert | Intended action. It is possible to emulate this ping signature using another ping utility. This kind of alert is unknown, but I think we should leave it on with mid priority tag. |
| Root cause | Generated manually |
| Percentage | 17% |
| Whois Command | Same IP causing trouble of most of the alerts, I recommend blocking that IP address 192.168.5.1 unless he is the network administrator |



```
 length = 56

000 : 63 45 A2 4D AC 16 08 00 08 09 0A 0B 0C 0D 0E 0F    cE.M............
010 : 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F    ................
020 : 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F     !"#$%&'()*+,-./
030 : 30 31 32 33 34 35 36 37                            01234567
```



| | | | |
|---|---|---|---|
| □ #0-(1-137) [local] [snort] ICMP PING *NIX2011-04-11 00:03:47 | 192.168.5.1 | 192.168.5.50 | ICMP |
| □ #1-(1-141) [local] [snort] ICMP PING *NIX2011-04-11 00:03:47 | 192.168.5.1 | 192.168.5.50 | ICMP |
| □ #2-(1-205) [local] [snort] ICMP PING *NIX2011-04-11 00:09:35 | 192.168.5.1 | 192.168.5.220 | ICMP |
| □ #3-(1-209) [local] [snort] ICMP PING *NIX2011-04-11 00:09:35 | 192.168.5.1 | 192.168.5.220 | ICMP |
| □ #4-(1-280) [local] [snort] ICMP PING *NIX2011-04-11 00:14:37 | 192.168.5.1 | 192.168.5.67 | ICMP |
| □ #5-(1-284) [local] [snort] ICMP PING *NIX2011-04-11 00:14:37 | 192.168.5.1 | 192.168.5.67 | ICMP |
| □ #6-(1-316) [local] [snort] ICMP PING *NIX2011-04-11 00:17:08 | 192.168.5.1 | 192.168.5.108 | ICMP |
| □ #7-(1-358) [local] [snort] ICMP PING *NIX2011-04-11 00:27:29 | 192.168.5.1 | 192.168.5.59 | ICMP |
| □ #8-(1-369) [local] [snort] ICMP PING *NIX2011-04-11 00:29:09 | 192.168.5.1 | 192.168.5.215 | ICMP |
| □ #9-(1-449) [local] [snort] ICMP PING *NIX2011-04-11 00:36:46 | 192.168.5.1 | 192.168.5.139 | ICMP |
| □ #10-(1-455)[local] [snort] ICMP PING *NIX2011-04-11 00:37:40 | 192.168.5.1 | 192.168.5.188 | ICMP |
| □ #11-(1-459)[local] [snort] ICMP PING *NIX2011-04-11 00:37:40 | 192.168.5.1 | 192.168.5.188 | ICMP |

Filter: (ip.addr eq 192.168.5.1 and ip.addr eq 192.16 Expression... Clear Apply

| No. | Time | Source | Destination | Protocol | Info |
|---|---|---|---|---|---|
| 156078 | 2011-04-10 19:09:34.984535 | 192.168.5.1 | 192.168.5.220 | DNS | Standard query response A 69.210.231.17 |

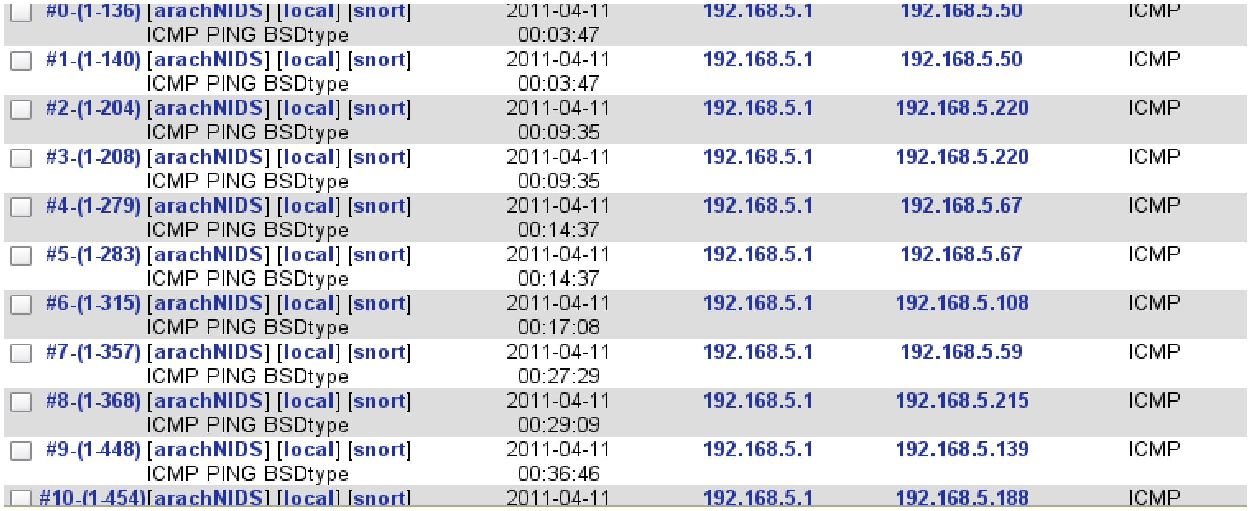
```
▷ Frame 156078: 214 bytes on wire (1712 bits), 214 bytes captured (1712 bits)
▷ Ethernet II, Src: Portwell_17:c2:de (00:90:fb:17:c2:de), Dst: Rim_f3:b5:eb (cc:55:ad:f3:b5:eb)
▷ Internet Protocol, Src: 192.168.5.1 (192.168.5.1), Dst: 192.168.5.220 (192.168.5.220)
▷ User Datagram Protocol, Src Port: domain (53), Dst Port: 61780 (61780)
▽ Domain Name System (response)
    Transaction ID: 0x0cea
  ▷ Flags: 0x8180 (Standard query response, No error)
    Questions: 1
    Answer RRs: 1
    Authority RRs: 3
    Additional RRs: 3
  ▷ Queries
  ▽ Answers
    ▷ nmd.bn02904.chcgil.wayport.net: type A, class IN, addr 69.210.231.17
  ▽ Authoritative nameservers
    ▷ bn02904.chcgil.wayport.net: type NS, class IN, ns auth0.wayport.net
    ▷ bn02904.chcgil.wayport.net: type NS, class IN, ns auth1.wayport.net
    ▷ bn02904.chcgil.wayport.net: type NS, class IN, ns auth2.wayport.net
  ▽ Additional records
    ▷ auth0.wayport.net: type A, class IN, addr 216.12.255.1

0020  05 dc 00 35 f1 54 00 b4  54 0c 0c ea 81 80 00 01   ...5.T.. T.......
0030  00 01 00 03 00 03 03 6e  6d 64 07 62 6e 30 32 39   .......n md.bn029
0040  30 34 06 63 68 63 67 69  6c 07 77 61 79 70 6f 72   04.chcgi l.waypor
0050  74 03 6e 65 74 00 00 01  00 01 c0 0c 00 01 00 01   t.net... ........
```

Figure 8: ICMP PING OR *NIX PING screenshots

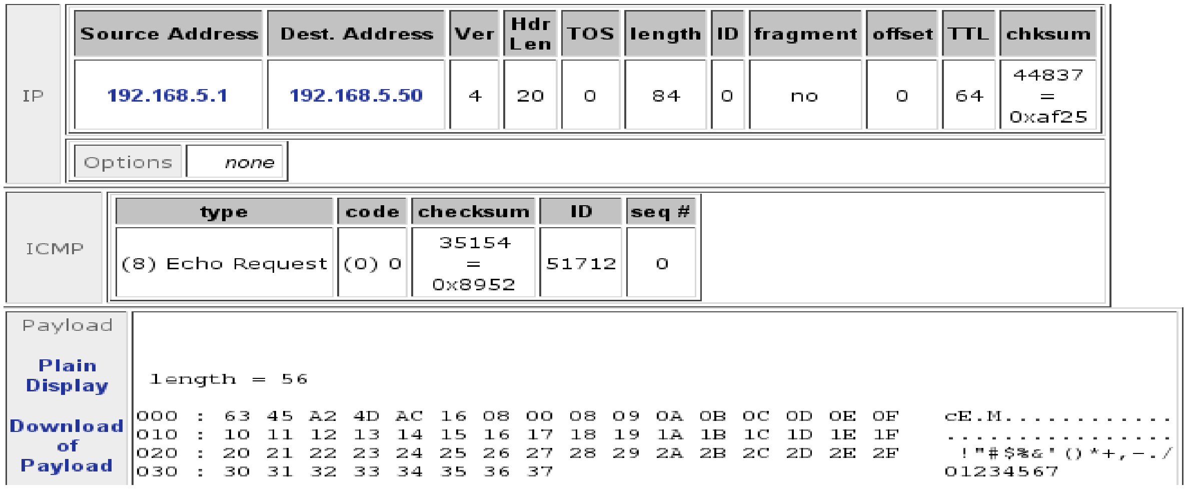
### 3- ICMP PING BSD type

An ICMP echo request is made from a Berkeley Systems Development (BSD) host. Therefore, an ICMP echo request is used by the ping command to elicit an ICMP echo reply from a listening live host. An echo request that originates from a host running a BSD TCP/IP networking stack such as FreeBSD, NetBSD, or OpenBSD, will contain a unique payload in the message request. An attacker may attempt to determine live hosts in a network prior to launching an attack. [9]

| ICMP Ping BSD type | Analysis |
|---|---|
| Kind of Alert | Mostly false positive "noise". Network administrator may use an ICMP echo request to legitimately troubleshoot networking problems. |
| Root cause | DNS cache |
| Percentage | 3% 13 alerts went off |
| Whois Command | Private addresses requesting UDP/ DNS |



| | | | | | |
|---|---|---|---|---|---|
| □ | #0-(1-136) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:03:47 | 192.168.5.1 | 192.168.5.50 | ICMP |
| □ | #1-(1-140) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:03:47 | 192.168.5.1 | 192.168.5.50 | ICMP |
| □ | #2-(1-204) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:09:35 | 192.168.5.1 | 192.168.5.220 | ICMP |
| □ | #3-(1-208) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:09:35 | 192.168.5.1 | 192.168.5.220 | ICMP |
| □ | #4-(1-279) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:14:37 | 192.168.5.1 | 192.168.5.67 | ICMP |
| □ | #5-(1-283) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:14:37 | 192.168.5.1 | 192.168.5.67 | ICMP |
| □ | #6-(1-315) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:17:08 | 192.168.5.1 | 192.168.5.108 | ICMP |
| □ | #7-(1-357) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:27:29 | 192.168.5.1 | 192.168.5.59 | ICMP |
| □ | #8-(1-368) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:29:09 | 192.168.5.1 | 192.168.5.215 | ICMP |
| □ | #9-(1-448) [arachNIDS] [local] [snort] ICMP PING BSDtype | 2011-04-11 00:36:46 | 192.168.5.1 | 192.168.5.139 | ICMP |
| □ | #10-(1-454)[arachNIDS] [local] [snort] | 2011-04-11 | 192.168.5.1 | 192.168.5.188 | ICMP |



| | Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 192.168.5.1 | 192.168.5.50 | 4 | 20 | 0 | 84 | 0 | no | 0 | 64 | 44837<br>=<br>0xaf25 |
| | Options | none | | | | | | | | | |

| | type | code | checksum | ID | seq # |
|---|---|---|---|---|---|
| ICMP | (8) Echo Request | (0) 0 | 35154<br>=<br>0x8952 | 51712 | 0 |

Payload

Plain Display

Download of Payload

```
 length = 56

000 : 63 45 A2 4D AC 16 08 00 08 09 0A 0B 0C 0D 0E 0F   cE.M............
010 : 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F   ................
020 : 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F    !"#$%&'()*+,-./
030 : 30 31 32 33 34 35 36 37                           01234567
```

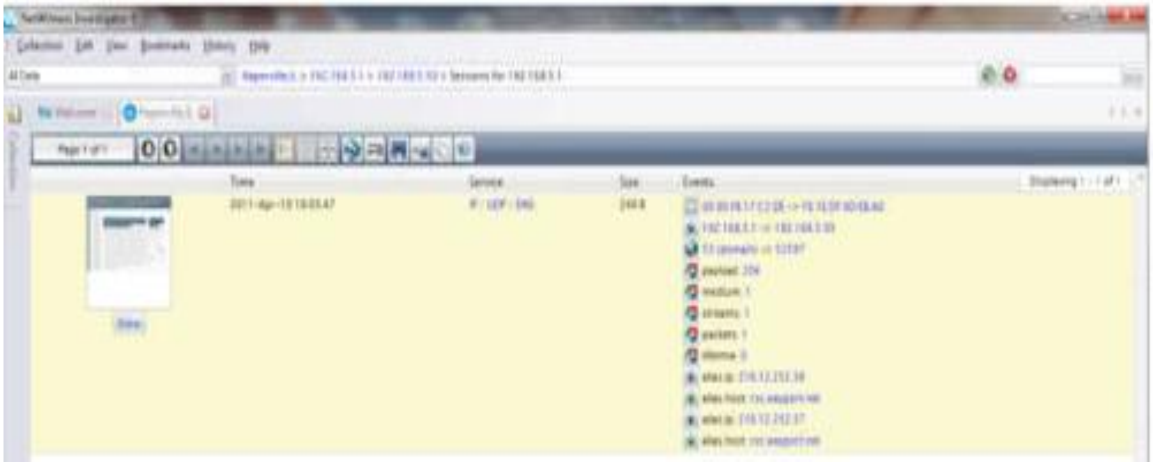
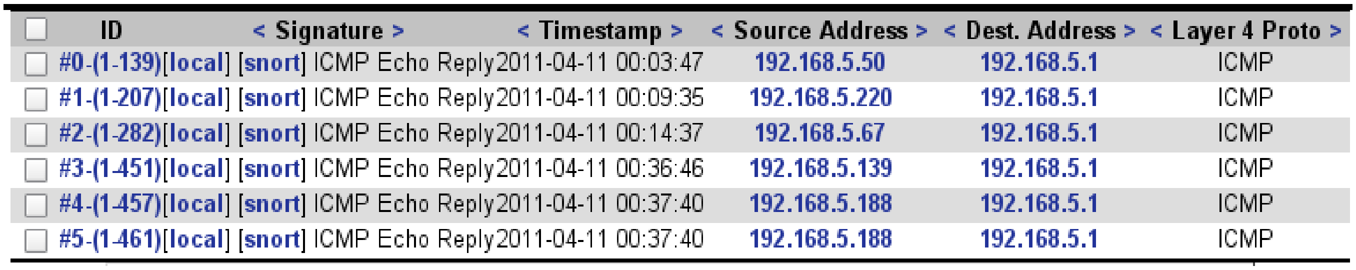


Figure 9: ICMP PING BSD type screenshots.

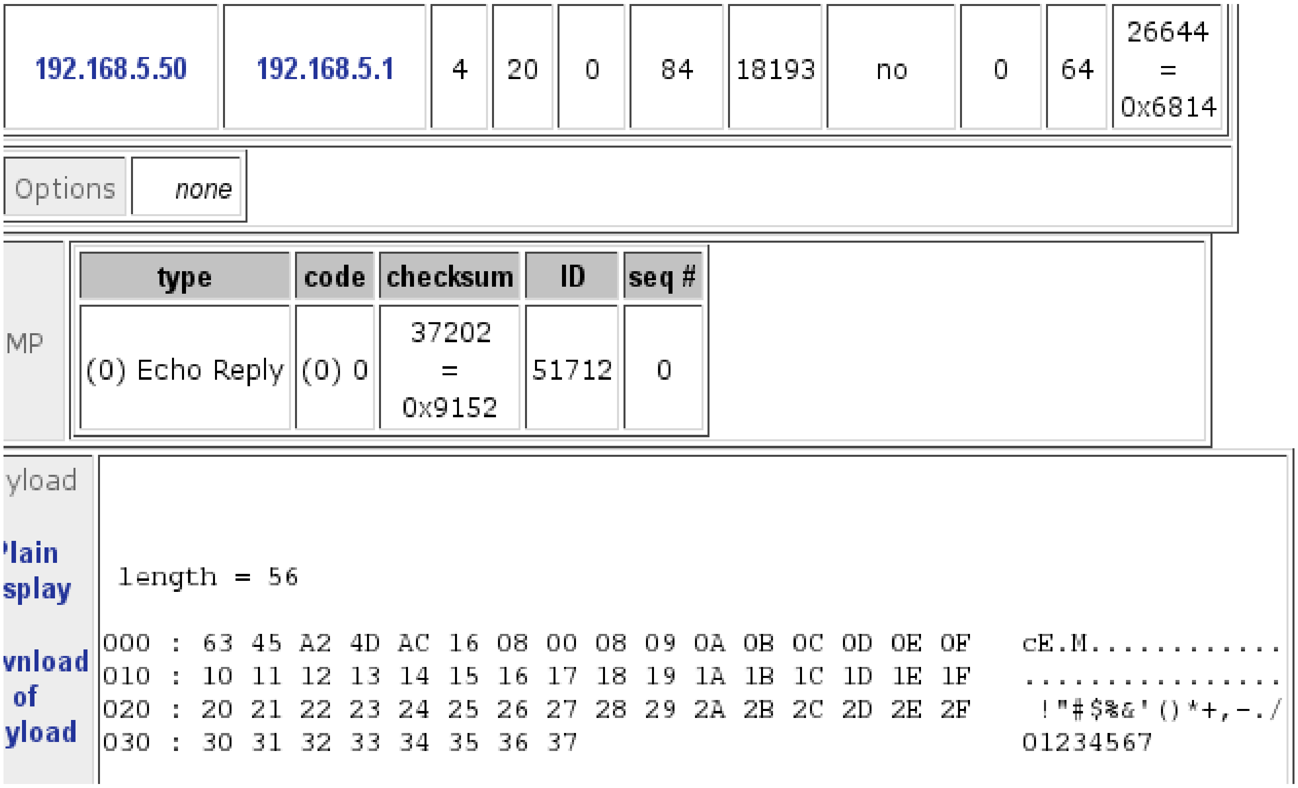
### 4- ICMP Echo Reply

This valuable information is generated when a network host generates an ICMP Echo Reply in response to an ICMP Echo Request message. An ICMP Echo Reply message is sent in response to an ICMP Echo Request message. If the ICMP Echo Reply message reaches the requesting host, it indicates that the replying host is alive. ICMP Type 0 Code 0 is the RFC defined messaging type for ICMP Echo Reply datagram. This type of message is used to determine if a host is active on the network. A remote attacker may use ICMP Echo Request datagram to determine active hosts on the network in prelude further attacks. [10]

| ICMP Echo Replay | Analysis |
|---|---|
| Kind of Alert | Serious, unless the administrator is testing the network. We should look at it as an actual attack, why and who they are testing port availability of the server. Blocking unneeded ports are necessary. |
| Root cause | Generated automatically |
| Percentage | 1% 2 alerts went off |
| Whois Command | Private IP 192.168.5.1 |



| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-139) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:03:47 | 192.168.5.50 | 192.168.5.1 | ICMP |
| □ | #1-(1-207) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:09:35 | 192.168.5.220 | 192.168.5.1 | ICMP |
| □ | #2-(1-282) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:14:37 | 192.168.5.67 | 192.168.5.1 | ICMP |
| □ | #3-(1-451) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:36:46 | 192.168.5.139 | 192.168.5.1 | ICMP |
| □ | #4-(1-457) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:37:40 | 192.168.5.188 | 192.168.5.1 | ICMP |
| □ | #5-(1-461) | [local] [snort] ICMP Echo Reply | 2011-04-11 00:37:40 | 192.168.5.188 | 192.168.5.1 | ICMP |



|  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| 192.168.5.50 | 192.168.5.1 | 4 | 20 | 0 | 84 | 18193 | no | 0 | 64 | 26644<br>=<br>0x6814 |

| Options | none |
|---|---|

| type | code | checksum | ID | seq # |
|---|---|---|---|---|
| (0) Echo Reply | (0) 0 | 37202<br>=<br>0x9152 | 51712 | 0 |

```
length = 56

000 : 63 45 A2 4D AC 16 08 00 08 09 0A 0B 0C 0D 0E 0F   cE.M............
010 : 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F   ................
020 : 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F    !"#$%&'()*+,-./
030 : 30 31 32 33 34 35 36 37                           01234567
```

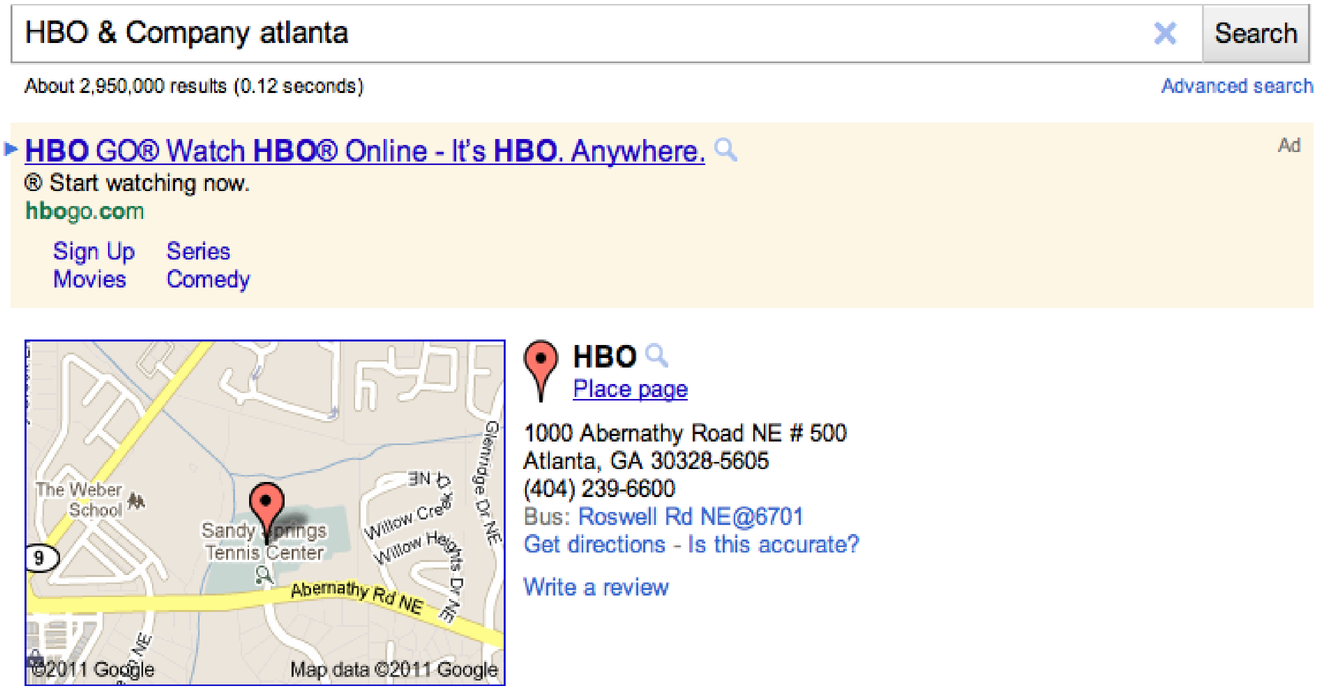
Figure 10: ICMP Echo Replay screenshots.

### 5- ICMP Time-To-Live Exceeded in Transit

Internet Control Message Protocol is part of the Internet Protocol Suite. ICMP messages are typically generated in response to errors in IP datagrams or for diagnostic or routing purposes. This incident is generated when a routing device detects that a packet has exceeded the maximum number of allowable hops during the packet flight. Each packet is assigned an initial Time To Live (TTL) value before being sent. This value is usually determined by the operating system of the given TCP/IP stack. The TTL value represents the maximum number of hops a packet may take before being expired and dropped by a routing device, some packets have the maximum allowable hops like HBO, and DIGI. This is done to banish lost or misguided packets from the network. The trace route utility assigns its own TTL values to dictate the number of hops a packet takes, to discover all the routing devices that are traversed by a packet.

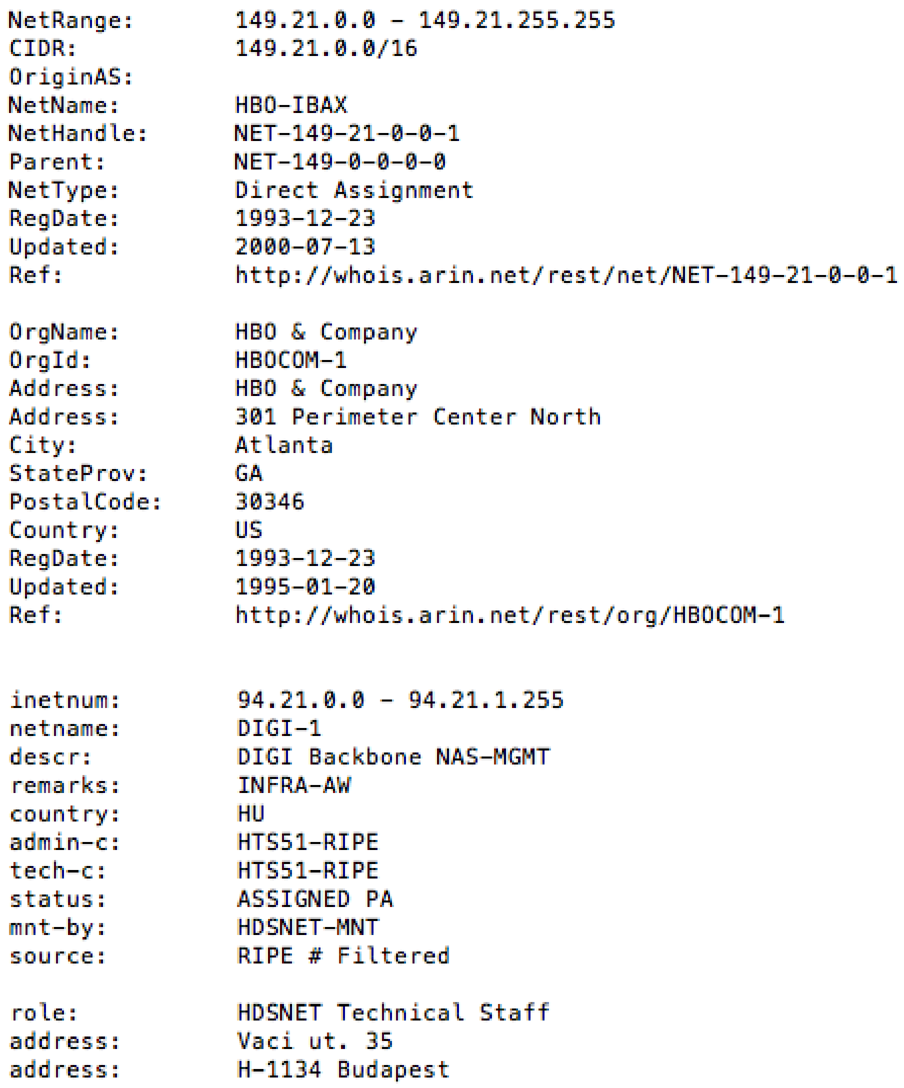
During the process, an ICMP "Time Exceeded in Transit" message may be observed. That is why ICMP traffic may be used to map a network, or help fingerprint an Operating system type, and version. If a router in your network sends this message, it may be an indication that an attacker is attempting a trace route of a host in your network and discover your network topology. [11]

| ICMP TTL Exceeded In Transit | Analysis |
|---|---|
| Kind of Alert | This kind of alert is false positive. Because an ICMP "Time Exceeded in Transit" message sent outbound if any inbound packet has exceeded the maximum allowable hops. |
| Root cause | Generated manually |
| Percentage | 1% 4 alerts went off |
| Whois Command | HBO & Company Atlanta |

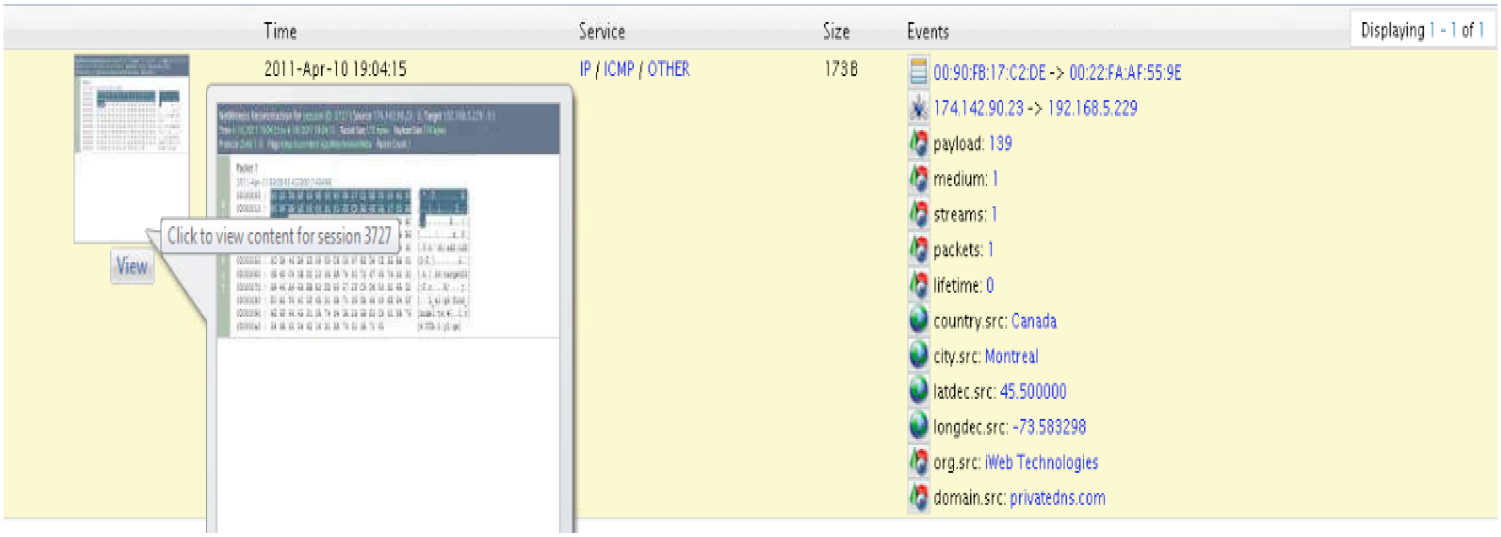


About 2,950,000 results (0.12 seconds)

HBO
Place page
1000 Abernathy Road NE # 500
Atlanta, GA 30328-5605
(404) 239-6600
Bus: Roswell Rd NE@6701
Get directions - Is this accurate?
Write a review

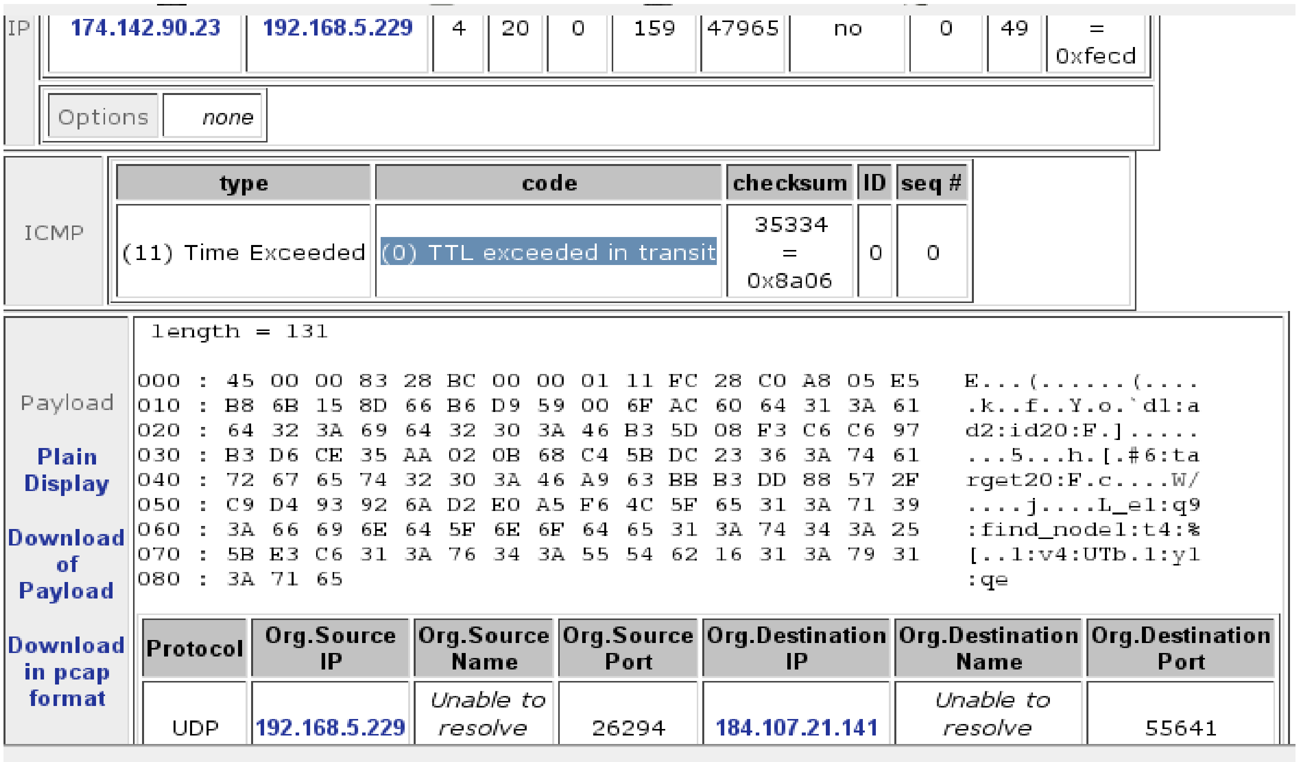


```
NetRange:       149.21.0.0 - 149.21.255.255
CIDR:           149.21.0.0/16
OriginAS:
NetName:        HBO-IBAX
NetHandle:      NET-149-21-0-0-1
Parent:         NET-149-0-0-0-0
NetType:        Direct Assignment
RegDate:        1993-12-23
Updated:        2000-07-13
Ref:            http://whois.arin.net/rest/net/NET-149-21-0-0-1

OrgName:        HBO & Company
OrgId:          HBOCOM-1
Address:        HBO & Company
Address:        301 Perimeter Center North
City:           Atlanta
StateProv:      GA
PostalCode:     30346
Country:        US
RegDate:        1993-12-23
Updated:        1995-01-20
Ref:            http://whois.arin.net/rest/org/HBOCOM-1


inetnum:        94.21.0.0 - 94.21.1.255
netname:        DIGI-1
descr:          DIGI Backbone NAS-MGMT
remarks:        INFRA-AW
country:        HU
admin-c:        HTS51-RIPE
tech-c:         HTS51-RIPE
status:         ASSIGNED PA
mnt-by:         HDSNET-MNT
source:         RIPE # Filtered

role:           HDSNET Technical Staff
address:        Vaci ut. 35
address:        H-1134 Budapest
```



Displaying 1 - 1 of 1

| Time | Service | Size | Events |
|---|---|---|---|
| 2011-Apr-10 19:04:15 | IP / ICMP / OTHER | 173 B | 00:90:FB:17:C2:DE -> 00:22:FA:AF:55:9E<br>174.142.90.23 -> 192.168.5.229<br>payload: 139<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0<br>country.src: Canada<br>city.src: Montreal<br>latdec.src: 45.500000<br>longdec.src: -73.583298<br>org.src: iWeb Technologies<br>domain.src: privatedns.com |



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 174.142.90.23 | 192.168.5.229 | 4 | 20 | 0 | 159 | 47965 | no | 0 | 49 | =<br>0xfecd |

| | |
|---|---|
| Options | none |

| type | code | checksum | ID | seq # |
|---|---|---|---|---|
| (11) Time Exceeded | (0) TTL exceeded in transit | 35334<br>=<br>0x8a06 | 0 | 0 |

```
 length = 131

000 : 45 00 00 83 28 BC 00 00 01 11 FC 28 C0 A8 05 E5   E...(......(....
010 : B8 6B 15 8D 66 B6 D9 59 00 6F AC 60 64 31 3A 61   .k..f..Y.o.`d1:a
020 : 64 32 3A 69 64 32 30 3A 46 B3 5D 08 F3 C6 C6 97   d2:id20:F.].....
030 : B3 D6 CE 35 AA 02 0B 68 C4 5B DC 23 36 3A 74 61   ...5...h.[.#6:ta
040 : 72 67 65 74 32 30 3A 46 A9 63 BB B3 DD 88 57 2F   rget20:F.c....W/
050 : C9 D4 93 92 6A D2 E0 A5 F6 4C 5F 65 31 3A 71 39   ....j....L_e1:q9
060 : 3A 66 69 6E 64 5F 6E 6F 64 65 31 3A 74 34 3A 25   :find_node1:t4:%
070 : 5B E3 C6 31 3A 76 34 3A 55 54 62 16 31 3A 79 31   [..1:v4:UTb.1:y1
080 : 3A 71 65                                          :qe
```

| Protocol | Org.Source IP | Org.Source Name | Org.Source Port | Org.Destination IP | Org.Destination Name | Org.Destination Port |
|---|---|---|---|---|---|---|
| UDP | 192.168.5.229 | Unable to resolve | 26294 | 184.107.21.141 | Unable to resolve | 55641 |

Figure 11: ICMP TTL Exceeded in Transit screenshots.

### 6- ICMP Destination Unreachable Network Unreachable

ICMP Network Unreachable datagram incident is detected on the network when the route to the destination network is not available. This could be an indication of routing problems on the network. This rule generates informational events about the network. Large numbers of these messages on the network could indication routing problems, faulty routing devices, or improperly configured hosts. However, this is not an attack at all, numerous tools and scripts can generate these types of ICMP datagram. [12]

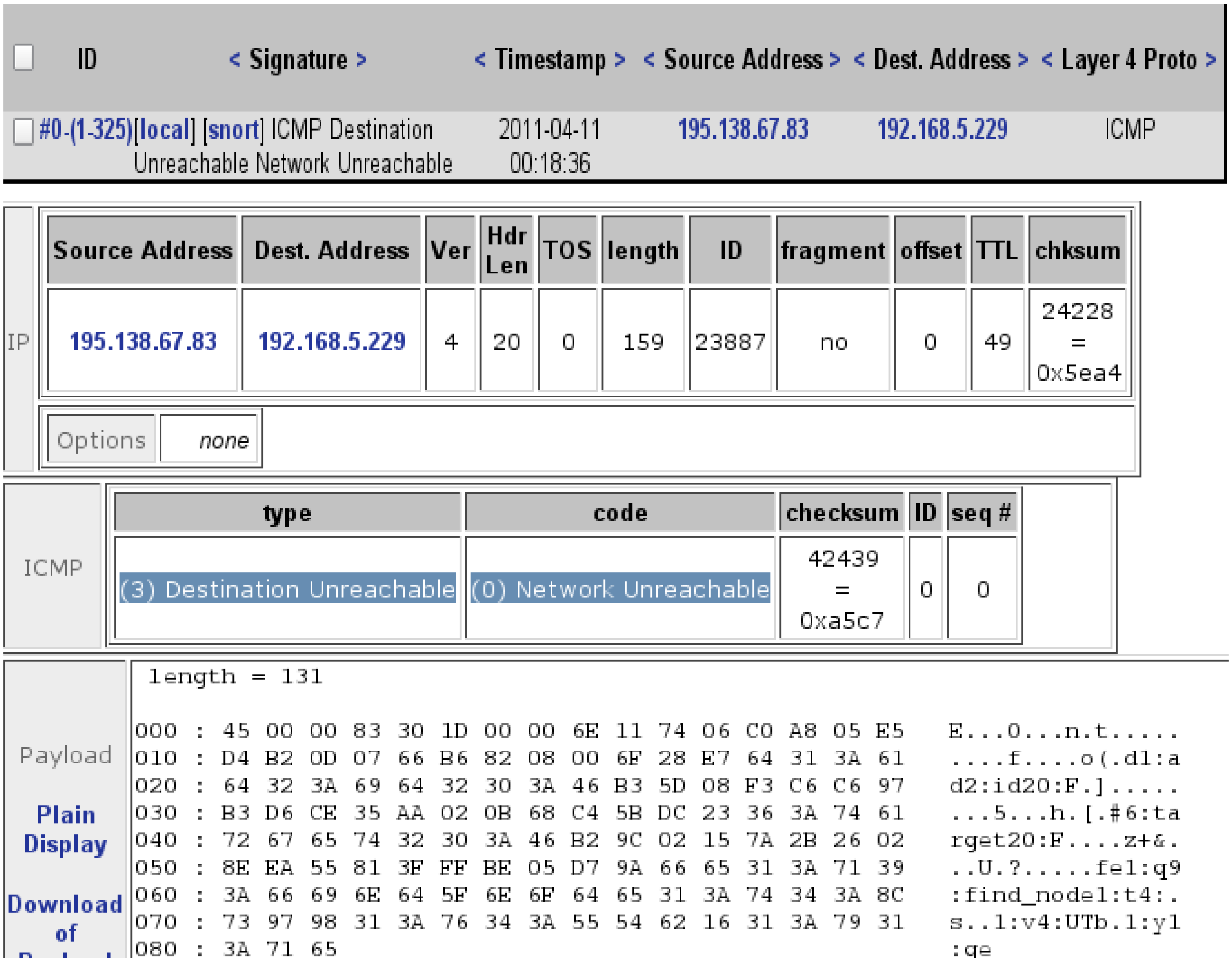
Network administrator should answer the following questions:

1- Is the specified destination address a valid network?

2- Is the link up from the router sending the Network Unreachable message?

3- Is the port in the router configured with the correct address mask value?

| ICMP Unreachable Network | Analysis |
|---|---|
| Kind of Alert | False Positive |
| Root cause | Automatically when the network has routing problems, faulty routing devices, or improperly configured hosts. |
| Percentage | 0% only one time |
| Whois Command | Both addresses “Source & Destination” are private |



| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-325) | [local] [snort] ICMP Destination Unreachable Network Unreachable | 2011-04-11 00:18:36 | 195.138.67.83 | 192.168.5.229 | ICMP |

| | Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 195.138.67.83 | 192.168.5.229 | 4 | 20 | 0 | 159 | 23887 | no | 0 | 49 | 24228<br>=<br>0x5ea4 |
| | Options | *none* | | | | | | | | | |

| | type | code | checksum | ID | seq # |
|---|---|---|---|---|---|
| ICMP | (3) Destination Unreachable | (0) Network Unreachable | 42439<br>=<br>0xa5c7 | 0 | 0 |

```
 length = 131

000 : 45 00 00 83 30 1D 00 00 6E 11 74 06 C0 A8 05 E5   E...0...n.t.....
010 : D4 B2 0D 07 66 B6 82 08 00 6F 28 E7 64 31 3A 61   ....f....o(.d1:a
020 : 64 32 3A 69 64 32 30 3A 46 B3 5D 08 F3 C6 C6 97   d2:id20:F.].....
030 : B3 D6 CE 35 AA 02 0B 68 C4 5B DC 23 36 3A 74 61   ...5...h.[.#6:ta
040 : 72 67 65 74 32 30 3A 46 B2 9C 02 15 7A 2B 26 02   rget20:F....z+&.
050 : 8E EA 55 81 3F FF BE 05 D7 9A 66 65 31 3A 71 39   ..U.?.....fe1:q9
060 : 3A 66 69 6E 64 5F 6E 6F 64 65 31 3A 74 34 3A 8C   :find_node1:t4:.
070 : 73 97 98 31 3A 76 34 3A 55 54 62 16 31 3A 79 31   s..1:v4:UTb.1:y1
080 : 3A 71 65                                          :qe
```

Figure 12: ICMP Unreachable Network screenshots.

### 7- ICMP Destination Unreachable Host

ICMP Destination unreachable host is generated when an ICMP Host Unreachable datagram is detected on the network. Routers will generate this message when the route to the destination host on a directly connected network is not available. This occurs when no ARP response is received from the destination network. As I mention before, this is not an attack and several tools and scripts can generate these types of ICMP unreachable host messages. [13]

Network administrator should answer the following questions:

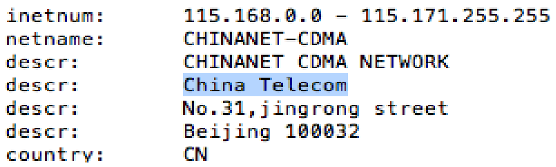
1- Are you assured that the intervening communications infrastructure is working properly?

2- Is the specified destination address the correct address for the host?

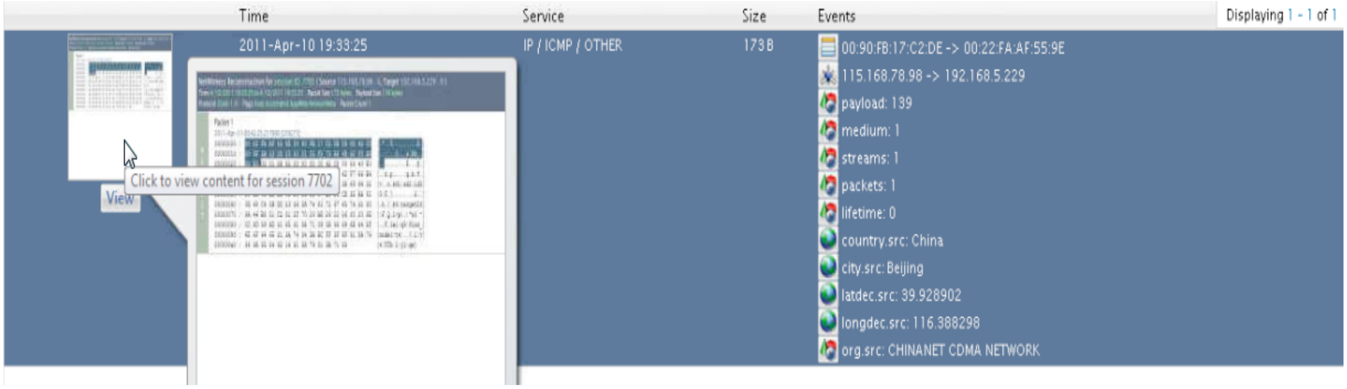
3- Is the host currently on-line and active?

4- Are there any physical problems on the destination network?

| ICMP Unreachable Host | Analysis |
|---|---|
| Kind of Alert | False Positive. |
| Root cause | Generated Automatically, indicates routing problems |
| Percentage | 2% |
| Whois Command | China Telecom, and other private IPs |



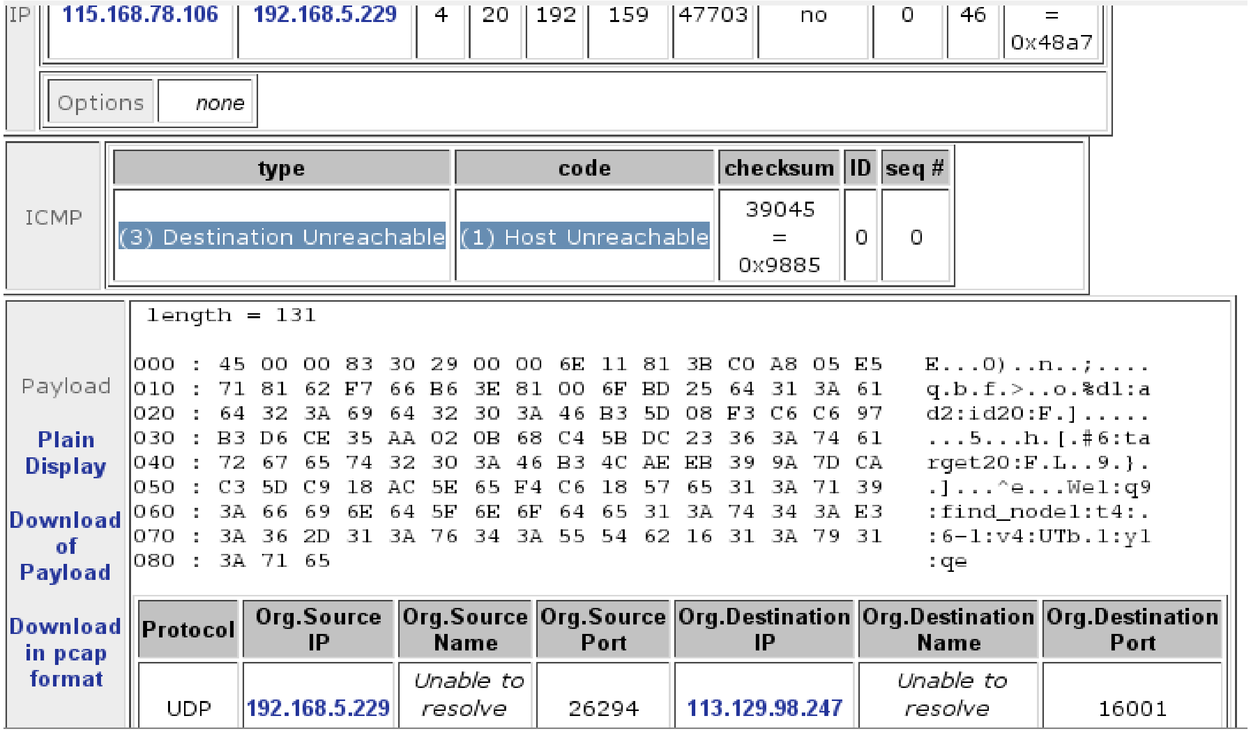
```
inetnum:        115.168.0.0 - 115.171.255.255
netname:        CHINANET-CDMA
descr:          CHINANET CDMA NETWORK
descr:          China Telecom
descr:          No.31,jingrong street
descr:          Beijing 100032
country:        CN
```



Displaying 1 - 1 of 1

| Time | Service | Size | Events |
|---|---|---|---|
| 2011-Apr-10 19:33:25 | IP / ICMP / OTHER | 173 B | 00:90:FB:17:C2:DE -> 00:22:FA:AF:55:9E<br>115.168.78.98 -> 192.168.5.229<br>payload: 139<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0<br>country.src: China<br>city.src: Beijing<br>latdec.src: 39.928902<br>longdec.src: 116.388298<br>org.src: CHINANET CDMA NETWORK |

Click to view content for session 7702



IP

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 115.168.78.106 | 192.168.5.229 | 4 | 20 | 192 | 159 | 47703 | no | 0 | 46 | =<br>0x48a7 |

| | |
|---|---|
| Options | *none* |

ICMP

| type | code | checksum | ID | seq # |
|---|---|---|---|---|
| (3) Destination Unreachable | (1) Host Unreachable | 39045<br>=<br>0x9885 | 0 | 0 |

Payload

```
 length = 131

000 :  45 00 00 83 30 29 00 00 6E 11 81 3B C0 A8 05 E5   E...0)..n..;....
010 :  71 81 62 F7 66 B6 3E 81 00 6F BD 25 64 31 3A 61   q.b.f.>..o.%d1:a
020 :  64 32 3A 69 64 32 30 3A 46 B3 5D 08 F3 C6 C6 97   d2:id20:F.].....
030 :  B3 D6 CE 35 AA 02 0B 68 C4 5B DC 23 36 3A 74 61   ...5...h.[.#6:ta
040 :  72 67 65 74 32 30 3A 46 B3 4C AE EB 39 9A 7D CA   rget20:F.L..9.}.
050 :  C3 5D C9 18 AC 5E 65 F4 C6 18 57 65 31 3A 71 39   .]...^e...We1:q9
060 :  3A 66 69 6E 64 5F 6E 6F 64 65 31 3A 74 34 3A E3   :find_node1:t4:.
070 :  3A 36 2D 31 3A 76 34 3A 55 54 62 16 31 3A 79 31   :6-1:v4:UTb.1:y1
080 :  3A 71 65                                          :qe
```

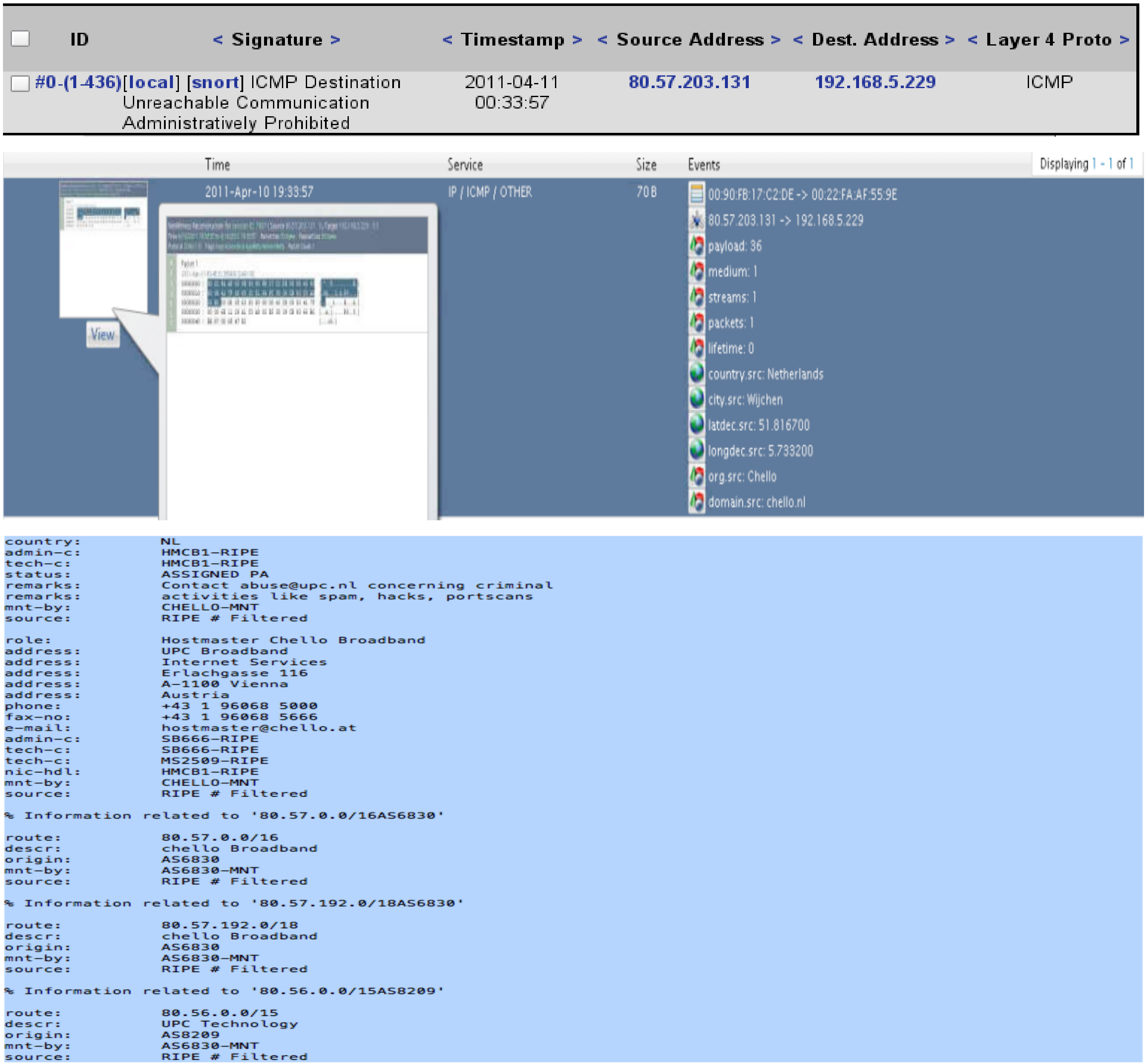
| Protocol | Org.Source IP | Org.Source Name | Org.Source Port | Org.Destination IP | Org.Destination Name | Org.Destination Port |
|---|---|---|---|---|---|---|
| UDP | 192.168.5.229 | *Unable to resolve* | 26294 | 113.129.98.247 | *Unable to resolve* | 16001 |

Figure 13: ICMP unreachable host screenshots.

### 8- DELETED ICMP Unreachable Communication Administratively Prohibited

This occurs in a point where is a router was unable to forward a packet due to filtering and used the Internet Control Message Protocol to alert involved hosts. A packet sent between two points on a network was administratively prohibited via filtering of some sort. The host or device performing the filtering returned an ICMP message informing the apparent source host that filtering had been done. This particular message is meant only to be informative but can be indicative of malicious activity (spoofed traffic, or Denial of Service Attack). However, an attacker can use to spoof spoofed source addresses. If and when the traffic gets filtered and an ICMP message is returned, the spoofed source address will be the recipient of the ICMP message. A similar situation may occur when a large portscan is occurring and an attempt is made to mask the true source of the scan by using spoof source addresses by using tools are readily available that can craft arbitrary ICMP packets. It is also possible to spoof packets using arbitrary addresses potentially causing intermediary routers to generate ICMP messages. [14]

| ICMP administratively prohibited | Analysis |
|---|---|
| Kind of Alert | False Positive, unless excessive ICMP messages were found. |
| Root cause | Generated manually |
| Percentage | 0% only one time |
| Whois Command | Going to Netherland |



| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-436) | [local] [snort] ICMP Destination Unreachable Communication Administratively Prohibited | 2011-04-11 00:33:57 | 80.57.203.131 | 192.168.5.229 | ICMP |

| Time | Service | Size | Events |
|---|---|---|---|
| 2011-Apr-10 19:33:57 | IP / ICMP / OTHER | 70 B | 00:90:FB:17:C2:DE -> 00:22:FA:AF:55:9E<br>80.57.203.131 -> 192.168.5.229<br>payload: 36<br>medium: 1<br>streams: 1<br>packets: 1<br>lifetime: 0<br>country.src: Netherlands<br>city.src: Wijchen<br>latdec.src: 51.816700<br>longdec.src: 5.733200<br>org.src: Chello<br>domain.src: chello.nl |

```
country:        NL
admin-c:        HMCB1-RIPE
tech-c:         HMCB1-RIPE
status:         ASSIGNED PA
remarks:        Contact abuse@upc.nl concerning criminal
remarks:        activities like spam, hacks, portscans
mnt-by:         CHELLO-MNT
source:         RIPE # Filtered

role:           Hostmaster Chello Broadband
address:        UPC Broadband
address:        Internet Services
address:        Erlachgasse 116
address:        A-1100 Vienna
address:        Austria
phone:          +43 1 96068 5000
fax-no:         +43 1 96068 5666
e-mail:         hostmaster@chello.at
admin-c:        SB666-RIPE
tech-c:         SB666-RIPE
tech-c:         MS2509-RIPE
nic-hdl:        HMCB1-RIPE
mnt-by:         CHELLO-MNT
source:         RIPE # Filtered

% Information related to '80.57.0.0/16AS6830'

route:          80.57.0.0/16
descr:          chello Broadband
origin:         AS6830
mnt-by:         AS6830-MNT
source:         RIPE # Filtered

% Information related to '80.57.192.0/18AS6830'

route:          80.57.192.0/18
descr:          chello Broadband
origin:         AS6830
mnt-by:         AS6830-MNT
source:         RIPE # Filtered

% Information related to '80.56.0.0/15AS8209'

route:          80.56.0.0/15
descr:          UPC Technology
origin:         AS8209
mnt-by:         AS6830-MNT
source:         RIPE # Filtered
```

Figure 14: ICMP Communication Administratively Prohibited Threats screenshots.

## Recommendations of ICMP Cyber Threats

Figure 15 summarizes the recommended actions for each of the threats described in this chapter.

| ICMP Threats | Recommended Action |
|---|---|
| **Destination Port Unreachable** | Examining the activity of the recipient of this packet to see if the recipient was responsible for scanning other behaviors. |
| **PING BSD type** | Use a packet filtering firewall to block ICMP packets |
| **PING & PING *NIX** | It is possible to emulate this ping signature using another ping utility. So blocking inbound ICMP echo requests using a packet filtering firewall is important to avoid this kind of threat. |
| **Echo Reply** | Use ingress filtering to prevent ICMP Type 0 Code 8 messages from entering the network. Only the networking administrator is allowed for testing purposes, otherwise, we should look at it as an actual attack. Why and who would benefit of testing port availability? So blocking unneeded ports and report the incident to top management is an important step. |
| **Time-To-Live Exceeded in Transit** | Sites may elect to disable this ICMP message on the outbound interface to prevent releasing potentially valuable information to serve a reconnaissance attempt about the network topology. This incident occurs if any inbound packet has exceeded the maximum allowable hops, which indicates a lost packet or routing problems such as a routing loop. |
| **Destination Network/Host Unreachable** | This is not an attack, it is only detects informational network information, where there is no corrective action necessary. |
| **Communication Administratively Prohibited** | There are none needed unless messages become excessive or appear to be invalid. Determine what traffic caused this particular ICMP message to be generated and act accordingly by blocking the source of that message. |

Figure 15: Recommendations of ICMP Cyber Threats

# Chapter 4: Applications Cyber Threat

This section will analyze cyber threats related to various software applications that I collected at a local Starbucks branch. The first step in securing a server is securing the running services and applications on that server. Most commonly available servers operate on a general-purpose operating system. Administrators can avoid many security issues if the applications running on servers are configured properly. These are the applications threat I collected: shellcode x86, CHAT Yahoo Messenger File Request, WEB-PHP arbitrary command execution, and MSN messenger http link transmission. First of all, Shellcode x86 is a TCP traffic streams on any x86 server for a x86 Studeo 0 system-call instructions, which are common in buffer overflow exploits technique that are used by hackers. While Chat File request and MSN link transmission are indication that Yahoo or MSN are been used and that violates network policy or leads to jeopardizing the system. In addition, this section will talk about the WEB-PHP arbitrary command execution threats and what Pajax is [15]. As before, the following will be discussed for each threat:

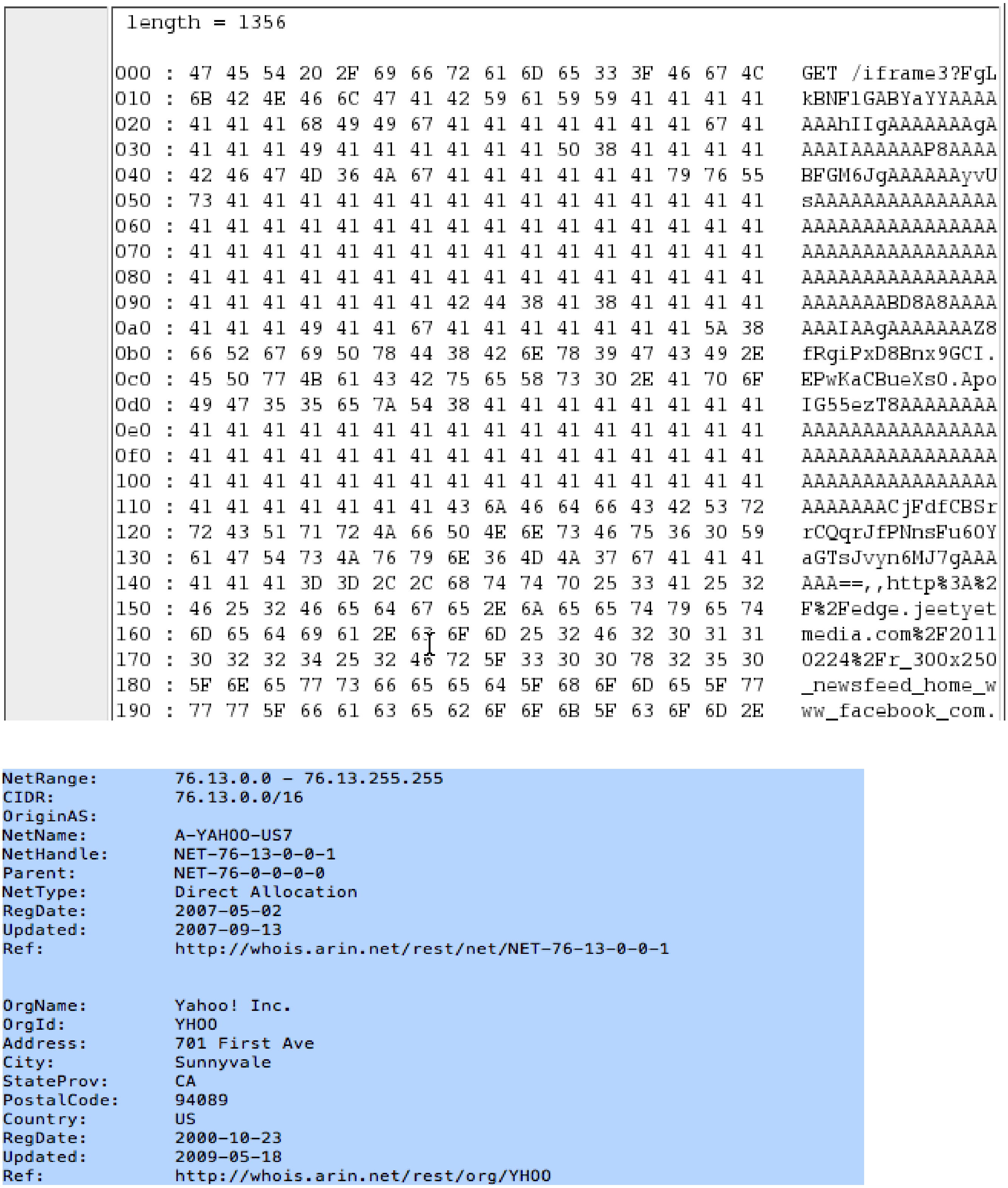
- The type of the threat
- Description of the threats
- The possible threat scenario
- The type of the alert
- The root cause of the incident
- Representative percentages
- If possible who initiated that threat and its recipients
- The recommended Action for administrator

### 1- SHELLCODE x86 inc ecx NOOP

As I explained above Shellcode is a TCP traffic streams on any x86 server for a x86 Studeo 0 system-call instructions, which are very common in buffer overflow exploits technique that are used by hackers. The name given to a class of assembly language programs that are used in the exploitation of a vulnerability using codes that executed in a shell. However, program execution flow is then manipulated so that the shellcode is executed. Shellcode often includes a call to the (0) function, which gives the super-user privileges. As an attacker could include shellcode, he/she would achieve this in a TCP packet being sent to a program with buffer overflow vulnerability. The existence of X86 binary assembler (0) instruction in TCP stream possibly indicates an attack intention. A remote attacker could be attempting to exploit buffer overflow vulnerability in a running program to gain full control over a system.

In particular, this is generated when an attempt is made to possibly overflow a buffer in memory. The NOOP warning occurs when a series of NOOP (no operation) are found in a stream. Most buffer overflow exploits typically use NOOPs sleds to pad the code. This rule detects a large number of consecutive NOOP instructions used in padding code. It's not specific to a particular service exploit, but rather used to try and detect buffer overflows in general. It is common for buffer overflow code to contain a large sequence of NOOP instructions as it increases the odds of successful execution of the useful shellcode. This might indicate someone is trying to use a buffer overflow exploit. Full compromise of system is possible if the exploit is successful. [16, 17]

| **SHELLCODE x86 inc NOOP** | **Analysis** |
|---|---|
| Kind of Alert | This is serious attack, where we should leave this alert with high priority tag, in case if generated by applications such as pdf or http when binary data is being transferred. This can lead to noise alerts sometimes when snort detects several (a) characters in a row - such as my screen shot shows 'aaaaaaaaaa'. |
| Root cause | Generated automatically |
| Percentage | 4% 20 alerts went off |
| Whois Command | Yahoo |



```
length = 1356

000 : 47 45 54 20 2F 69 66 72 61 6D 65 33 3F 46 67 4C   GET /iframe3?FgL
010 : 6B 42 4E 46 6C 47 41 42 59 61 59 59 41 41 41 41   kBNFlGABYaYYAAAA
020 : 41 41 41 68 49 49 67 41 41 41 41 41 41 41 67 41   AAAhIIgAAAAAAAgA
030 : 41 41 41 49 41 41 41 41 41 41 50 38 41 41 41 41   AAAIAAAAAAP8AAAA
040 : 42 46 47 4D 36 4A 67 41 41 41 41 41 41 79 76 55   BFGM6JgAAAAAAyvU
050 : 73 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   sAAAAAAAAAAAAAAA
060 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
070 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
080 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
090 : 41 41 41 41 41 41 41 42 44 38 41 38 41 41 41 41   AAAAAAABD8A8AAAA
0a0 : 41 41 41 49 41 41 67 41 41 41 41 41 41 41 5A 38   AAAIAAgAAAAAAAZ8
0b0 : 66 52 67 69 50 78 44 38 42 6E 78 39 47 43 49 2E   fRgiPxD8Bnx9GCI.
0c0 : 45 50 77 4B 61 43 42 75 65 58 73 30 2E 41 70 6F   EPwKaCBueXs0.Apo
0d0 : 49 47 35 35 65 7A 54 38 41 41 41 41 41 41 41 41   IG55ezT8AAAAAAAA
0e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
0f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
100 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
110 : 41 41 41 41 41 41 41 43 6A 46 64 66 43 42 53 72   AAAAAAACjFdfCBSr
120 : 72 43 51 71 72 4A 66 50 4E 6E 73 46 75 36 30 59   rCQqrJfPNnsFu60Y
130 : 61 47 54 73 4A 76 79 6E 36 4D 4A 37 67 41 41 41   aGTsJvyn6MJ7gAAA
140 : 41 41 41 3D 3D 2C 2C 68 74 74 70 25 33 41 25 32   AAA==,,http%3A%2
150 : 46 25 32 46 65 64 67 65 2E 6A 65 65 74 79 65 74   F%2Fedge.jeetyet
160 : 6D 65 64 69 61 2E 63 6F 6D 25 32 46 32 30 31 31   media.com%2F2011
170 : 30 32 32 34 25 32 46 72 5F 33 30 30 78 32 35 30   0224%2Fr_300x250
180 : 5F 6E 65 77 73 66 65 65 64 5F 68 6F 6D 65 5F 77   _newsfeed_home_w
190 : 77 77 5F 66 61 63 65 62 6F 6F 6B 5F 63 6F 6D 2E   ww_facebook_com.
```

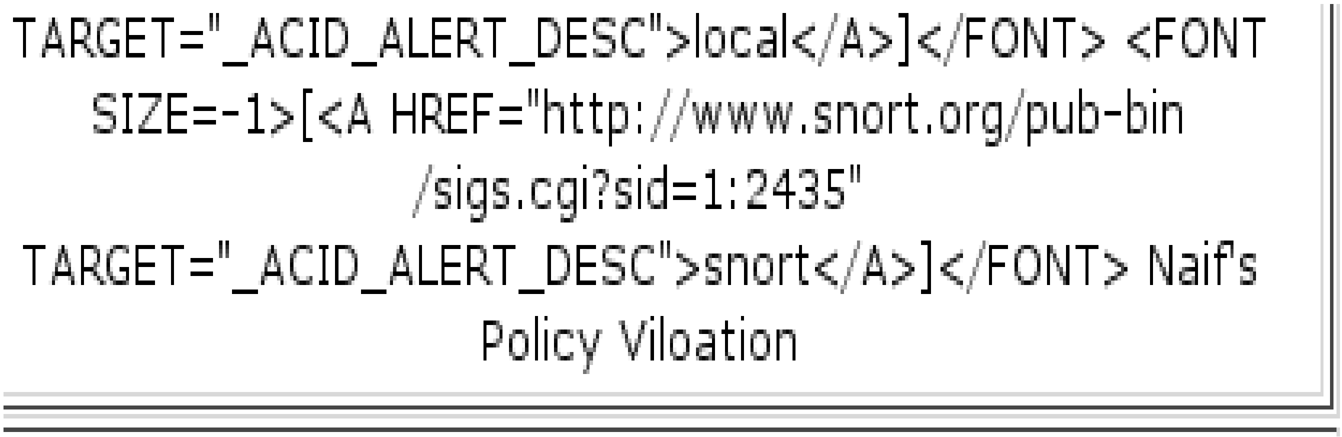
```
NetRange:       76.13.0.0 - 76.13.255.255
CIDR:           76.13.0.0/16
OriginAS:
NetName:        A-YAHOO-US7
NetHandle:      NET-76-13-0-0-1
Parent:         NET-76-0-0-0-0
NetType:        Direct Allocation
RegDate:        2007-05-02
Updated:        2007-09-13
Ref:            http://whois.arin.net/rest/net/NET-76-13-0-0-1


OrgName:        Yahoo! Inc.
OrgId:          YHOO
Address:        701 First Ave
City:           Sunnyvale
StateProv:      CA
PostalCode:     94089
Country:        US
RegDate:        2000-10-23
Updated:        2009-05-18
Ref:            http://whois.arin.net/rest/org/YHOO
```

Figure 16: Applications, Shell code x86 Threats screenshots.

**2- CHAT Yahoo Messenger File Transfer Initiation Request**

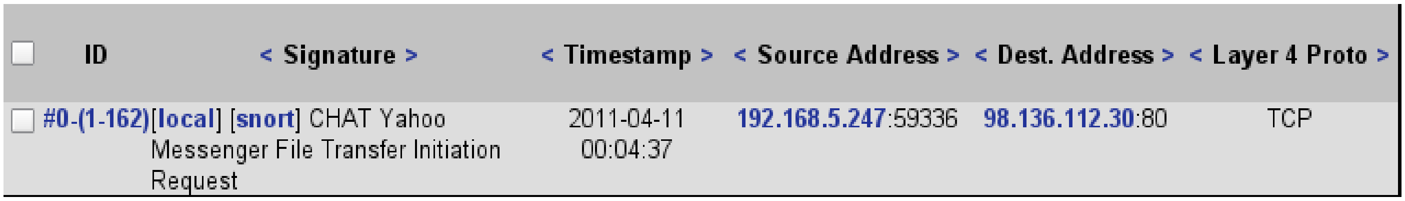
It occurs when network traffic that indicates an instant messaging client is being used. This event indicates that the Yahoo IM client is being used on the protected network. Specifically a Yahoo Messenger File Transfer Initiation Request was observed. It is possible to transfer files between hosts using instant messaging applications. This may lead to the loss of proprietary and confidential data. [18]

There is a simple rule to look for specific http requests. For example, a rule that looks for anyone going to type certain word like a black list! I chose my name “Naif” to trigger the alert by calling this alert Naif’ Policy violation.

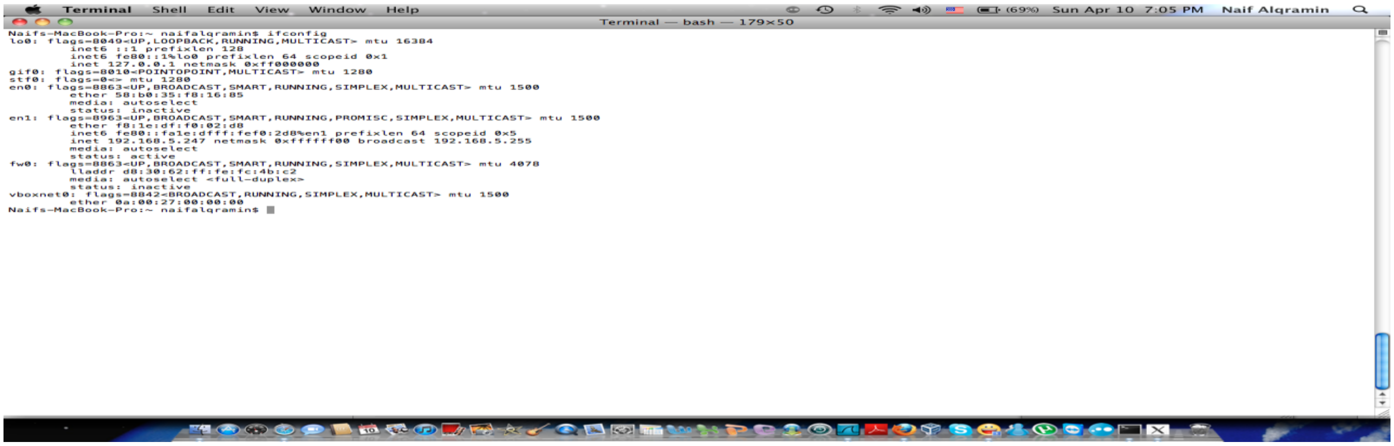


```
TARGET="_ACID_ALERT_DESC">local</A>]</FONT> <FONT
SIZE=-1>[<A HREF="http://www.snort.org/pub-bin
/sigs.cgi?sid=1:2435"
TARGET="_ACID_ALERT_DESC">snort</A>]</FONT> Naif's
Policy Viloation
```

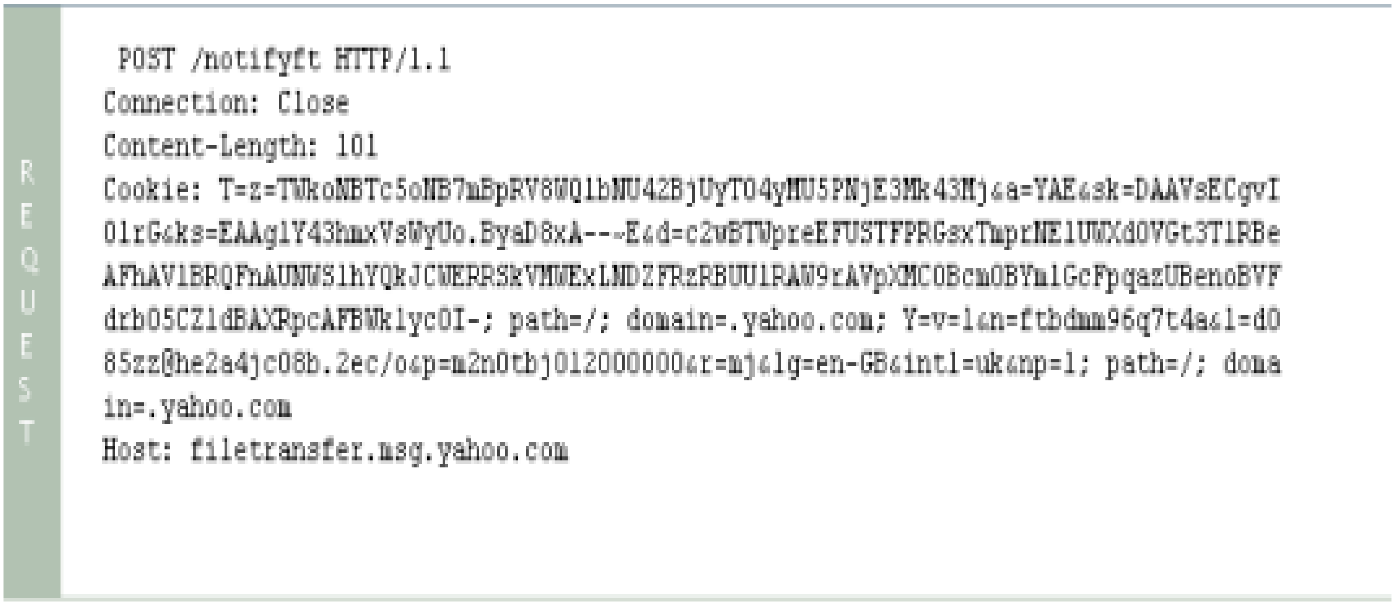
| Yahoo Messenger File Transfer | Analysis |
|---|---|
| Kind of Alert | True positive, it is useful to prevent business policy violation. |
| Root cause | Generated manually |
| Percentage | 0% only one time alert went off |
| Whois Command | Yahoo |



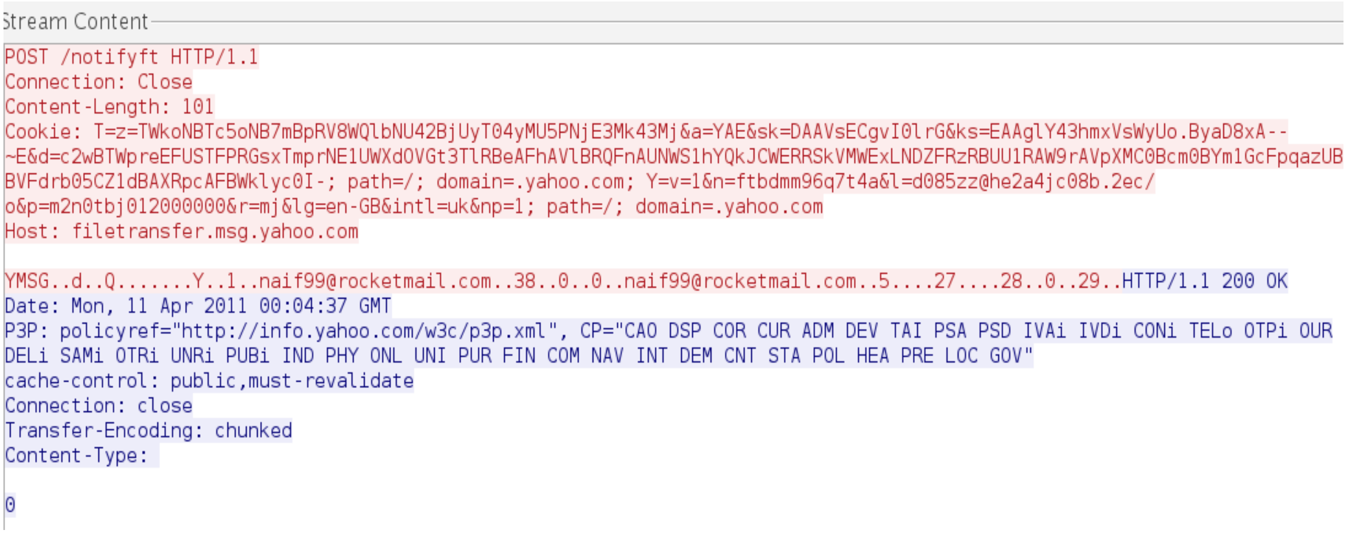
| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-162) | [local] [snort] CHAT Yahoo Messenger File Transfer Initiation Request | 2011-04-11 00:04:37 | 192.168.5.247:59336 | 98.136.112.30:80 | TCP |



```
Naifs-MacBook-Pro:~ naifalqramin$ ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        inet 127.0.0.1 netmask 0xff000000
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 58:b0:35:f8:16:85
        media: autoselect
        status: inactive
en1: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        ether f8:1e:df:f0:02:d8
        inet6 fe80::fa1e:dfff:fef0:2d8%en1 prefixlen 64 scopeid 0x5
        inet 192.168.5.247 netmask 0xffffff00 broadcast 192.168.5.255
        media: autoselect
        status: active
fw0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 4078
        lladdr d8:30:62:ff:fe:fc:4b:c2
        media: autoselect <full-duplex>
        status: inactive
vboxnet0: flags=8842<BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 0a:00:27:00:00:00
Naifs-MacBook-Pro:~ naifalqramin$
```



```
POST /notifyft HTTP/1.1
Connection: Close
Content-Length: 101
Cookie: T=z=TWkoNBTc5oNB7mBpRV8WQlbNU42BjUyT04yMU5PNjE3Mk43Mj&a=YAE&sk=DAAVsECgvI
0lrG&ks=EAAglY43hmxVsWyUo.ByaD8xA--~E&d=c2wBTWpreEFUSTFPRGsxTmprNE1UWXd0VGt3TlRBe
AFhAVlBRQFnAUNWS1hYQkJCWERRSkVMWExLNDZFRzRBUU1RAW9rAVpXMC0BcmOBYm1GcFpqazUBenoBVF
drb05CZ1dBAXRpcAFBWklyc0I-; path=/; domain=.yahoo.com; Y=v=1&n=ftbdmm96q7t4a&l=d0
85zz@he2a4jc08b.2ec/o&p=m2n0tbj012000000&r=mj&lg=en-GB&intl=uk&np=1; path=/; doma
in=.yahoo.com
Host: filetransfer.msg.yahoo.com
```



```
POST /notifyft HTTP/1.1
Connection: Close
Content-Length: 101
Cookie: T=z=TWkoNBTc5oNB7mBpRV8WQlbNU42BjUyT04yMU5PNjE3Mk43Mj&a=YAE&sk=DAAVsECgvI0lrG&ks=EAAglY43hmxVsWyUo.ByaD8xA--
~E&d=c2wBTWpreEFUSTFPRGsxTmprNE1UWXd0VGt3TlRBeAFhAVlBRQFnAUNWS1hYQkJCWERRSkVMWExLNDZFRzRBUU1RAW9rAVpXMC0Bcm0BYm1GcFpqazUB
BVFdrb05CZ1dBAXRpcAFBWklyc0I-; path=/; domain=.yahoo.com; Y=v=1&n=ftbdmm96q7t4a&l=d085zz@he2a4jc08b.2ec/
o&p=m2n0tbj012000000&r=mj&lg=en-GB&intl=uk&np=1; path=/; domain=.yahoo.com
Host: filetransfer.msg.yahoo.com
```

```
YMSG..d..Q.......Y..1..naif99@rocketmail.com..38..0..0..naif99@rocketmail.com..5....27....28..0..29..HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 00:04:37 GMT
P3P: policyref="http://info.yahoo.com/w3c/p3p.xml", CP="CAO DSP COR CUR ADM DEV TAI PSA PSD IVAi IVDi CONi TELo OTPi OUR
DELi SAMi OTRi UNRi PUBi IND PHY ONL UNI PUR FIN COM NAV INT DEM CNT STA POL HEA PRE LOC GOV"
cache-control: public,must-revalidate
Connection: close
Transfer-Encoding: chunked
Content-Type: 

0
```

Figure 17: Applications, Yahoo Messenger File Transfer Request Threats screenshots.

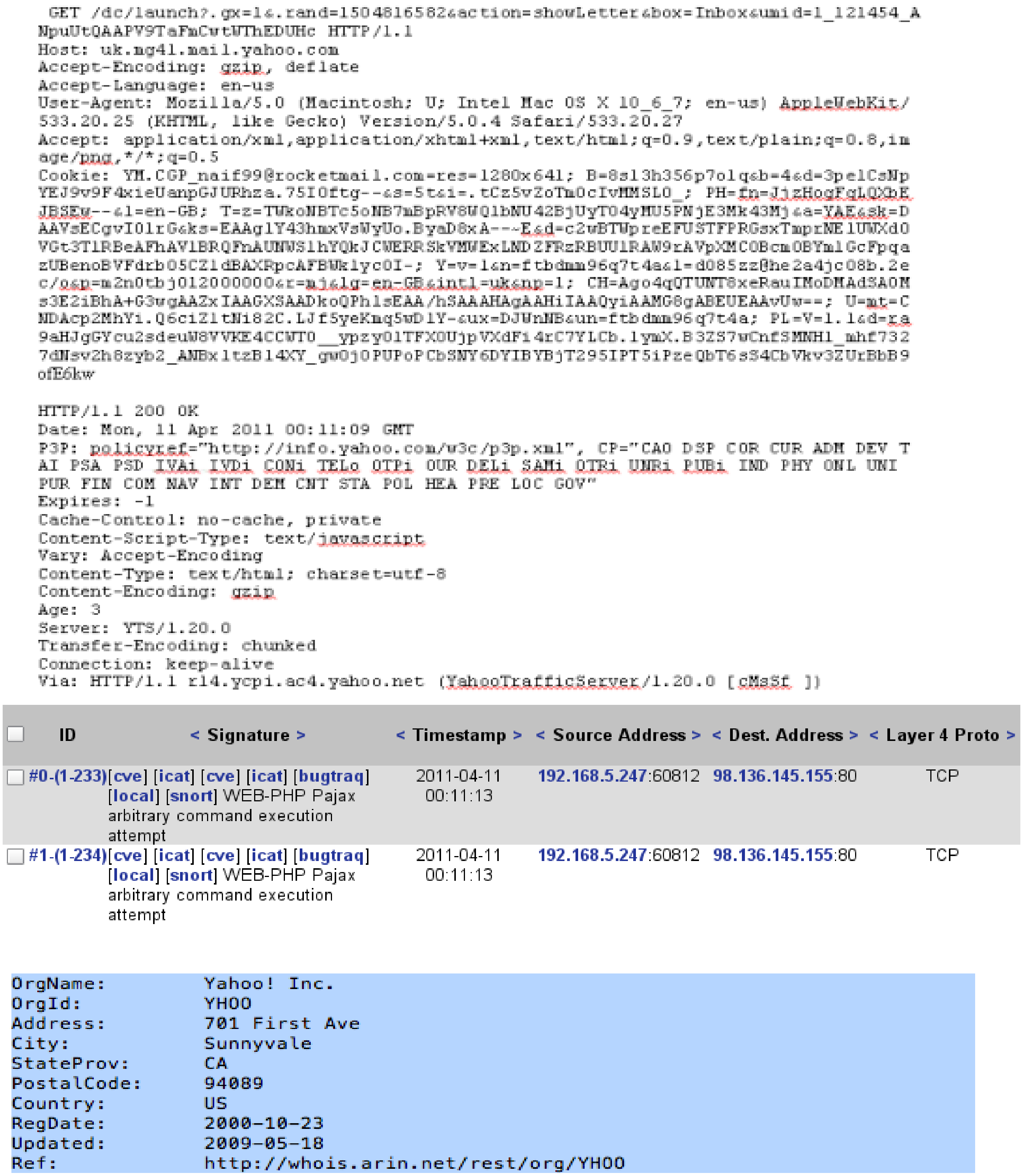
### 3- WEB-PHP Pajax arbitrary command execution attempt

First of all, Pajax is an AJAX framework, which allows simple PHP objects to be made remotely callable from within JavaScript, using XML Http Request. PAJAX utilizes an Object Request Broker (ORB) pattern allowing JavaScript objects to call methods of remote PHP objects via some remote interface. By using Pajax it is possible to write web applications that utilize PHP classes running on a remote server to perform operations. Pajax is able to create a remote JavaScript interface object and a stub on the server for some PHP class. The JavaScript interface communicates with the stub on the server, which invokes the called methods on the remote object.

However, this is made to exploit command injection vulnerability in the Pajax using CGI application running on a web server. This event indicates that an attempt has been made to inject a command from a remote machine in the Pajax application running on a web server. If stringent input checks are not configured properly by the CGI application, it may also be possible for an attacker to compromise the host running the application. The hacker may be able to execute system binaries or malicious code of their choosing. This event is generated when an attempt is made to gain unauthorized access to a CGI application running on a web server. Some applications do not perform stringent checks when validating the credentials of a client host connecting to the services offered on a host server. This can lead to unauthorized access and possibly escalated privileges to an administrator. Data stored on the machine can be compromised and the attacker can exploit trusted relationships between the victim server and other hosts as impacts. An attacker can inject commands to the application if user input is not correctly sanitized or checked before passing that input to the database. [19,20]

| WEB-PHP Pajax arbitrary command execution attempt | Analysis |
|---|---|
| Kind of Alert | Serious |
| Root cause | Manually |
| Percentage | %0 2 times only |
| Whois Command | Private address |

IP address 98.136.145.155 attacks Yahoo account. See figure 17 for attack scenario.



```
 GET /dc/launch?.gx=1&.rand=1504816582&action=showLetter&box=Inbox&umid=1_121454_A
NpuUtQAAPV9TaFmCwtWThEDUHc HTTP/1.1
Host: uk.mg41.mail.yahoo.com
Accept-Encoding: gzip, deflate
Accept-Language: en-us
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_7; en-us) AppleWebKit/
533.20.25 (KHTML, like Gecko) Version/5.0.4 Safari/533.20.27
Accept: application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,im
age/png,*/*;q=0.5
Cookie: YM.CGP_naif99@rocketmail.com=res=1280x641; B=8sl3h356p7olq&b=4&d=3pelCsNp
YEJ9v9F4xieUanpGJURhza.75I0ftg--&s=5t&i=.tCz5vZoTm0cIvMMSL0_; PH=fn=JjzHogFqLQXbE
JBSEw--&l=en-GB; T=z=TWkoNBTc5oNB7mBpRV8WQlbNU42BjUyTO4yMU5PNjE3Mk43Mj&a=YAE&sk=D
AAVsECgvI0lrG&ks=EAAglY43hmxVsWyUo.ByaD8xA--~E&d=c2wBTWpreEFUSTFPRGsxTmprNE1UWXd0
VGt3T1RBeAFhAVlBRQFnAUNWSlhYQkJCWERRSkVMWExLND2FRzRBUU1RAW9rAVpXMC0BcmOBYmlGcFpqa
zUBenoBVFdrb05CZldBAXRpcAFBWklyc0I-; Y=v=1&n=ftbdmm96q7t4a&l=d085zz@he2a4jc08b.2e
c/o&p=m2n0tbj012000000&r=mj&lg=en-GB&intl=uk&np=1; CH=Ago4qQTUNT8xeRauIMoDMAdSA0M
s3E2iBhA+G3wgAAZxIAAGXSAADkoQPhlsEAA/hSAAAHAgAAHiIAAQyiAAMG8gABEUEAAvUw==; U=mt=C
NDAcp2MhYi.Q6ciZltNi82C.LJf5yeKmq5wDlY-&ux=DJWnNB&un=ftbdmm96q7t4a; PL=V=1.1&d=ra
9aHJgGYcu2sdeuW8VVKE4CCWT0__ypzy0lTFX0UjpVXdFi4rC7YLCb.lymX.B3ZS7wCnfSMNH1_mhf732
7dNsv2h8zyb2_ANBxltzBl4XY_gw0j0PUPoPCbSNY6DYIBYBjT295IPT5iPzeQbT6sS4CbVkv3ZUrBbB9
ofE6kw

HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 00:11:09 GMT
P3P: policyref="http://info.yahoo.com/w3c/p3p.xml", CP="CAO DSP COR CUR ADM DEV T
AI PSA PSD IVAi IVDi CONi TELo OTPi OUR DELi SAMi OTRi UNRi PUBi IND PHY ONL UNI
PUR FIN COM NAV INT DEM CNT STA POL HEA PRE LOC GOV"
Expires: -1
Cache-Control: no-cache, private
Content-Script-Type: text/javascript
Vary: Accept-Encoding
Content-Type: text/html; charset=utf-8
Content-Encoding: gzip
Age: 3
Server: YTS/1.20.0
Transfer-Encoding: chunked
Connection: keep-alive
Via: HTTP/1.1 r14.ycpi.ac4.yahoo.net (YahooTrafficServer/1.20.0 [cMsSf ])
```

| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-233) | [cve] [icat] [cve] [icat] [bugtraq] [local] [snort] WEB-PHP Pajax arbitrary command execution attempt | 2011-04-11 00:11:13 | 192.168.5.247:60812 | 98.136.145.155:80 | TCP |
| □ | #1-(1-234) | [cve] [icat] [cve] [icat] [bugtraq] [local] [snort] WEB-PHP Pajax arbitrary command execution attempt | 2011-04-11 00:11:13 | 192.168.5.247:60812 | 98.136.145.155:80 | TCP |

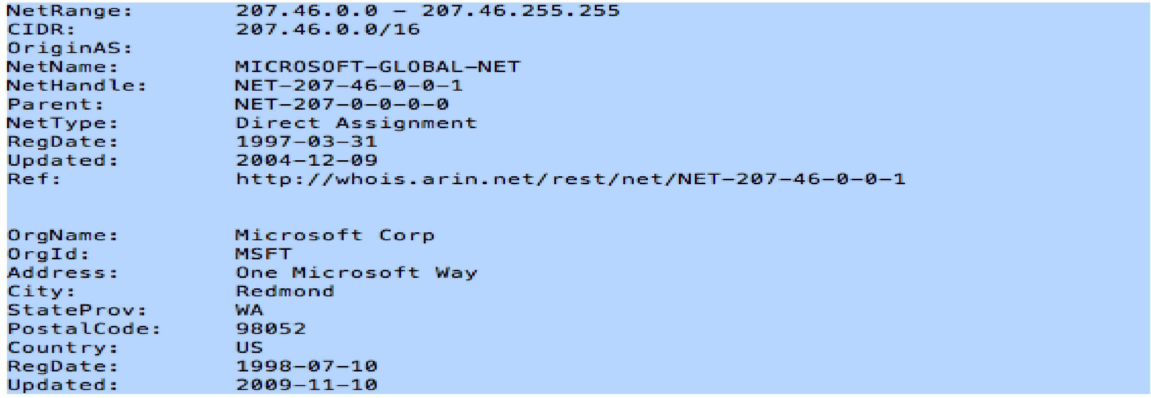
```
OrgName:        Yahoo! Inc.
OrgId:          YHOO
Address:        701 First Ave
City:           Sunnyvale
StateProv:      CA
PostalCode:     94089
Country:        US
RegDate:        2000-10-23
Updated:        2009-05-18
Ref:            http://whois.arin.net/rest/org/YHOO
```

Figure 18: WEB-PHP Pajax arbitrary command execution Threat screenshots.

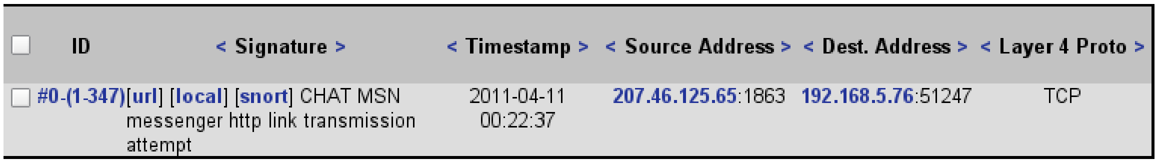
### 4- CHAT MSN messenger http link transmission attempt

This event is generated when network traffic that indicates MSN messenger is being used. Possible policy violation like if the use of MSN messenger may be prohibited by corporate policy in some network environments. This event indicates that the MSN messenger is being used on the protected network. [21,22]

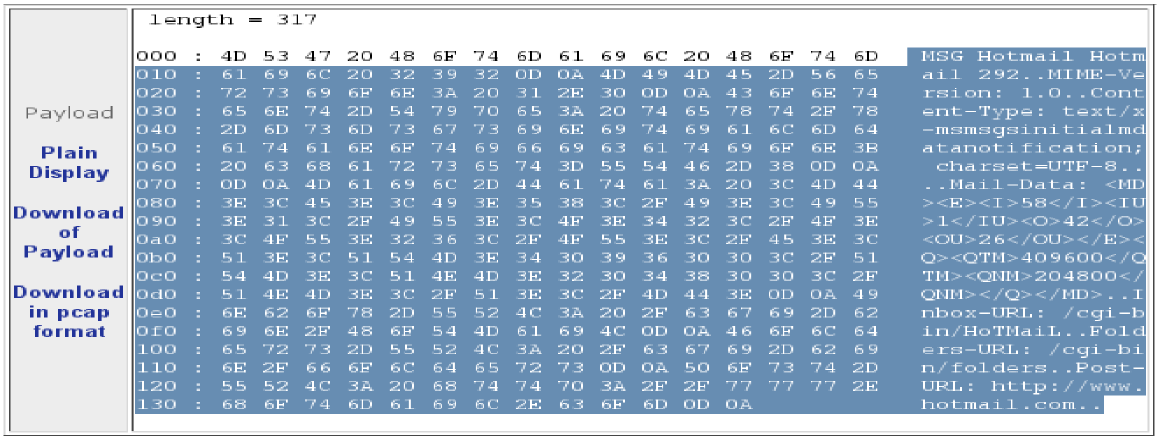
| MSN HTTP link transmission attempt | Analysis |
|---|---|
| Kind of Alert | True positive but not serious, it is useful to prevent business policy violation. |
| Root cause | Generated manually |
| Percentage | 0% one time |
| Whois Command | Microsoft Corporation |



```
NetRange:       207.46.0.0 - 207.46.255.255
CIDR:           207.46.0.0/16
OriginAS:
NetName:        MICROSOFT-GLOBAL-NET
NetHandle:      NET-207-46-0-0-1
Parent:         NET-207-0-0-0-0
NetType:        Direct Assignment
RegDate:        1997-03-31
Updated:        2004-12-09
Ref:            http://whois.arin.net/rest/net/NET-207-46-0-0-1


OrgName:        Microsoft Corp
OrgId:          MSFT
Address:        One Microsoft Way
City:           Redmond
StateProv:      WA
PostalCode:     98052
Country:        US
RegDate:        1998-07-10
Updated:        2009-11-10
```



| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-347) | [url] [local] [snort] CHAT MSN messenger http link transmission attempt | 2011-04-11 00:22:37 | 207.46.125.65:1863 | 192.168.5.76:51247 | TCP |

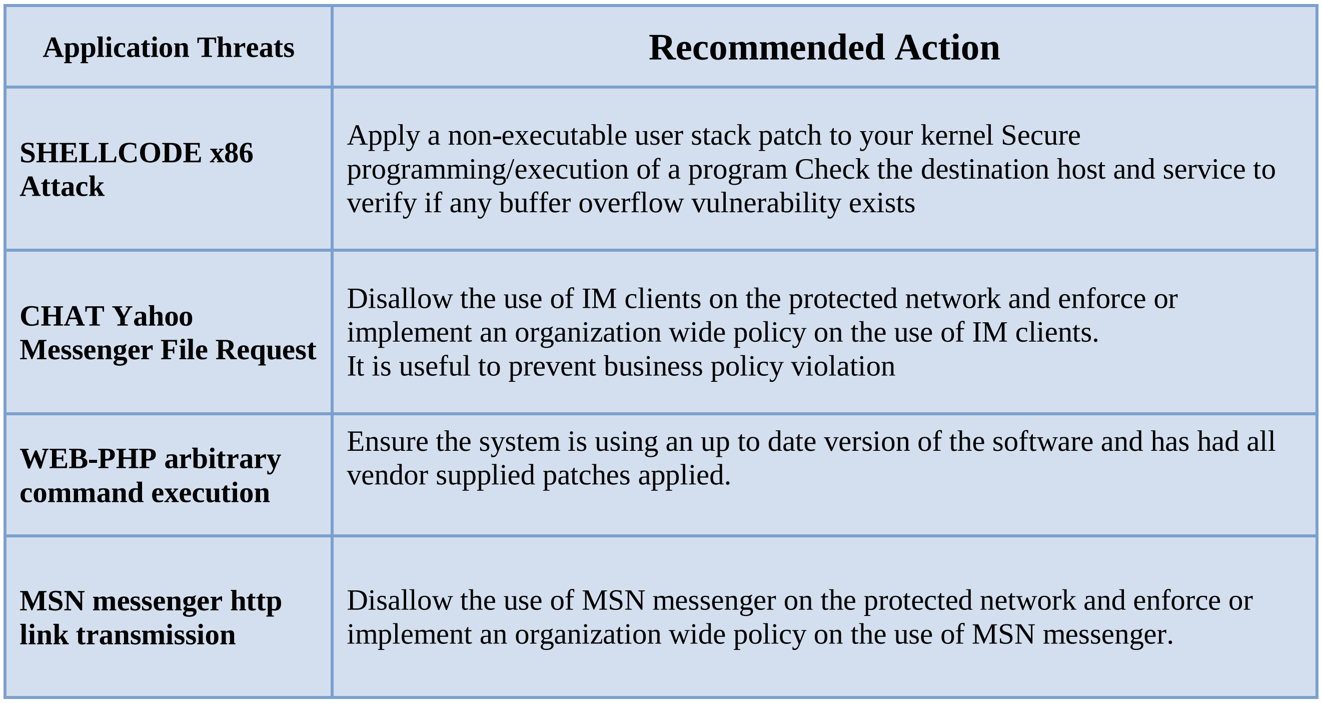


```
length = 317

000 : 4D 53 47 20 48 6F 74 6D 61 69 6C 20 48 6F 74 6D   MSG Hotmail Hotm
010 : 61 69 6C 20 32 39 32 0D 0A 4D 49 4D 45 2D 56 65   ail 292..MIME-Ve
020 : 72 73 69 6F 6E 3A 20 31 2E 30 0D 0A 43 6F 6E 74   rsion: 1.0..Cont
030 : 65 6E 74 2D 54 79 70 65 3A 20 74 65 78 74 2F 78   ent-Type: text/x
040 : 2D 6D 73 6D 73 67 73 69 6E 69 74 69 61 6C 6D 64   -msmsgsinitialmd
050 : 61 74 61 6E 6F 74 69 66 69 63 61 74 69 6F 6E 3B   atanotification;
060 : 20 63 68 61 72 73 65 74 3D 55 54 46 2D 38 0D 0A    charset=UTF-8..
070 : 0D 0A 4D 61 69 6C 2D 44 61 74 61 3A 20 3C 4D 44   ..Mail-Data: <MD
080 : 3E 3C 45 3E 3C 49 3E 35 38 3C 2F 49 3E 3C 49 55   ><E><I>58</I><IU
090 : 3E 31 3C 2F 49 55 3E 3C 4F 3E 34 32 3C 2F 4F 3E   >1</IU><O>42</O>
0a0 : 3C 4F 55 3E 32 36 3C 2F 4F 55 3E 3C 2F 45 3E 3C   <OU>26</OU></E><
0b0 : 51 3E 3C 51 54 4D 3E 34 30 39 36 30 30 3C 2F 51   Q><QTM>409600</Q
0c0 : 54 4D 3E 3C 51 4E 4D 3E 32 30 34 38 30 30 3C 2F   TM><QNM>204800</
0d0 : 51 4E 4D 3E 3C 2F 51 3E 3C 2F 4D 44 3E 0D 0A 49   QNM></Q></MD>..I
0e0 : 6E 62 6F 78 2D 55 52 4C 3A 20 2F 63 67 69 2D 62   nbox-URL: /cgi-b
0f0 : 69 6E 2F 48 6F 54 4D 61 69 4C 0D 0A 46 6F 6C 64   in/HoTMaiL..Fold
100 : 65 72 73 2D 55 52 4C 3A 20 2F 63 67 69 2D 62 69   ers-URL: /cgi-bi
110 : 6E 2F 66 6F 6C 64 65 72 73 0D 0A 50 6F 73 74 2D   n/folders..Post-
120 : 55 52 4C 3A 20 68 74 74 70 3A 2F 2F 77 77 77 2E   URL: http://www.
130 : 68 6F 74 6D 61 69 6C 2E 63 6F 6D 0D 0A            hotmail.com..
```

Figure 19: CHAT Messenger http link transmission Threat screenshots.

## Recommendations of Applications Cyber Threats

Figure 20 summarizes the recommended actions for each of the threats described in this chapter.



| Application Threats | Recommended Action |
|---|---|
| **SHELLCODE x86 Attack** | Apply a non-executable user stack patch to your kernel Secure programming/execution of a program Check the destination host and service to verify if any buffer overflow vulnerability exists |
| **CHAT Yahoo Messenger File Request** | Disallow the use of IM clients on the protected network and enforce or implement an organization wide policy on the use of IM clients.<br>It is useful to prevent business policy violation |
| **WEB-PHP arbitrary command execution** | Ensure the system is using an up to date version of the software and has had all vendor supplied patches applied. |
| **MSN messenger http link transmission** | Disallow the use of MSN messenger on the protected network and enforce or implement an organization wide policy on the use of MSN messenger. |

Figure 20: Recommendations of Applications Cyber Threats.

## Chapter 5: Web Cyber Threats:

This chapter focuses on web-related cyber threats observed during the listening session at Starbucks, which is all about Web HTTP and Web applications handler threats. In general, HTTP handler is the process, often called endpoint that runs in response to a request made to an ASP.NET Web application. ASP.NET is the most common page handler that processes .aspx files. When users request an .aspx file, the request is processed by the page through the page handler. These HTTP handlers can be created by custom output to the browser.

Therefore, an HTTP module is an assembly that is called on every request that is made to an application. As HTTP modules examine incoming and outgoing requests and takes action based on the request, and proper HTTP configuration also can be customized. Incoming requests can be examined, an HTTP module can perform custom authentication or other security checks before the requested page, XML Web service, or handler is called.

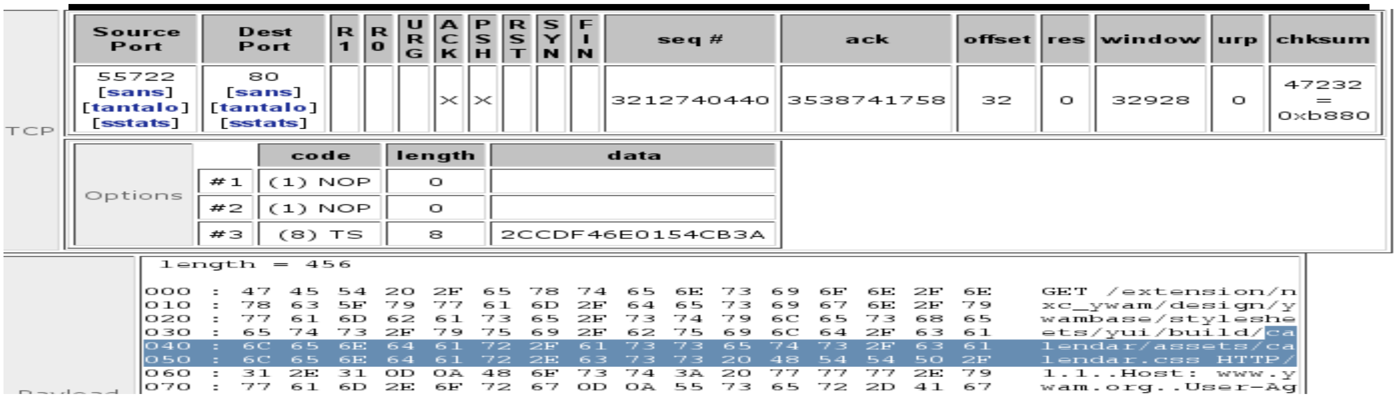
In this section particularly, I will go over the following threats; HTTP-Inspect Double Decoding Attack, WEB-CGI calendar access, WEB-CGI icat access, Open SSL get shared ciphers overflow attempt, IIS Unicode CODEPOINT Encoding, WEB-MISC handler access, and Oversize Chunk Encoding Attempt. [23,24] As before, the following will be discussed for each threat:

- The type of the threat
- Description of the threats
- The possible threat scenario
- The type of the alert
- The root cause of the incident
- The representative percentages
- If possible who initiated that threat and its recipients
- The recommended Action for administrator

### 1- WEB-CGI calendar access

This attempt is made to access a web application that may lead to exploitation of the application. An open source calendar perl script by Matt Kruse, allows commands to be executed without input verification using the perl open() function. ie /cgi bin/calendar_admin.pl place the string "|ping 127.0.0.1|" in the configuration file field, this executes the command "ping 127.0.0.1". An unauthenticated user can execute arbitrary programs on the server by accessing calendar_admin.pl and inputting commands such as "|mail /etc/passwd|" into the configuration file field. If your web server has pages by the name of calendar* this rule will fire often. Many sites now use calendar applications and this rule may generate a large number of false positives, it does not distinguish between perl cgi applications and php scripts because of purely written rules that need to be tuned. Consider tuning this rule for your site if it is generating a large number of false positives. If you use a calendar application, consider changing the name of the script to something other than "calendar". [25,26]

| Web-CGi Calendar Access | Analysis |
|---|---|
| Kind of Alert | This particular one is false positive, but I recommend setting rule to distinguish between them. We should rewrite the rule so that some of the rules cut a pretty wide swath so you may need to reduce their scope through some pass rules. |
| Root cause | Generated automatically |
| Percentage | 1% 3 alerts trigged |
| Whois Command | Going to NXC International SA. Switzerland. |



| Source Port | Dest Port | R 1 | R 0 | URG | ACK | PSH | RST | SYN | FIN | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 55722<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | X | | | | 3212740440 | 3538741758 | 32 | 0 | 32928 | 0 | 47232<br>=<br>0xb880 |

| Options | | code | length | data |
|---|---|---|---|---|
| | #1 | (1) NOP | 0 | |
| | #2 | (1) NOP | 0 | |
| | #3 | (8) TS | 8 | 2CCDF46E0154CB3A |

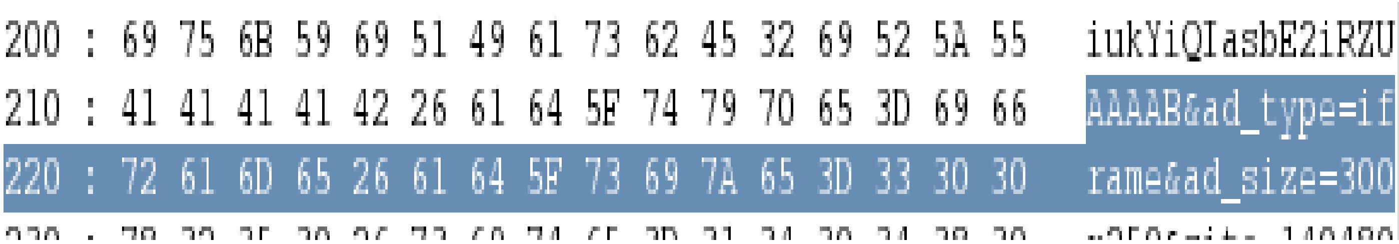
```
length = 456

000 : 47 45 54 20 2F 65 78 74 65 6E 73 69 6F 6E 2F 6E    GET /extension/n
010 : 78 63 5F 79 77 61 6D 2F 64 65 73 69 67 6E 2F 79    xc_ywam/design/y
020 : 77 61 6D 62 61 73 65 2F 73 74 79 6C 65 73 68 65    wambase/styleshe
030 : 65 74 73 2F 79 75 69 2F 62 75 69 6C 64 2F 63 61    ets/yui/build/ca
040 : 6C 65 6E 64 61 72 2F 61 73 73 65 74 73 2F 63 61    lendar/assets/ca
050 : 6C 65 6E 64 61 72 2E 63 73 73 20 48 54 54 50 2F    lendar.css HTTP/
060 : 31 2E 31 0D 0A 48 6F 73 74 3A 20 77 77 77 2E 79    1.1..Host: www.y
070 : 77 61 6D 2E 6F 72 67 0D 0A 55 73 65 72 2D 41 67    wam.org..User-Ag
```

Figure 21: Web Calendar Access Attack Screenshots.

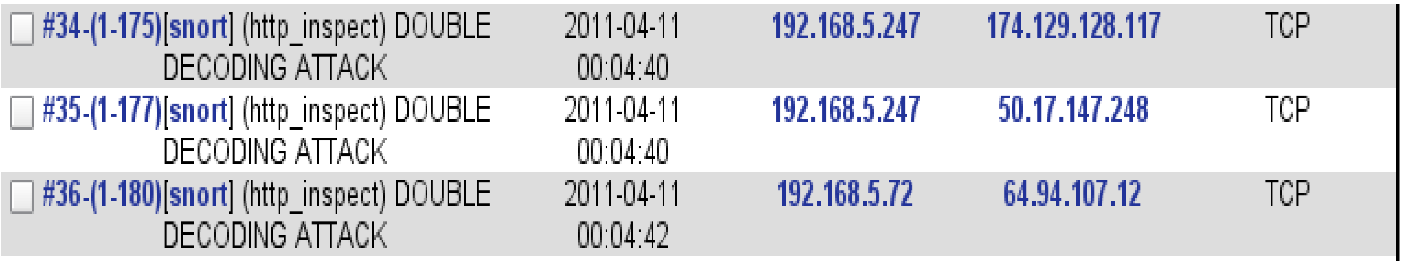
### 2- HTTP-Inspect Double Decoding Attack

This event is generated when double encoded characters are detected in web traffic. This is abnormal behavior and may be an indicator of a possible attack against a vulnerable system. This may also be an attempt to often evade IDS on Microsoft IIS Servers environment. Since IIS server has some vulnerabilities that can be exploited by HTTP Double decoding attack. An attacker might double encode the request to the web server, this may then evade an IDS monitoring traffic and could then launch a successful attack without being detected. [27]

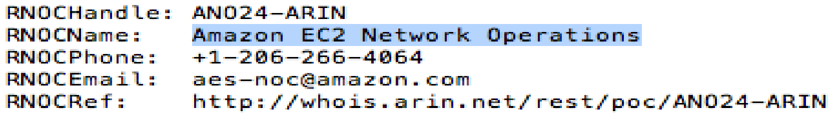
| HTTP-Inspect Double Decoding Attack | Analysis |
|---|---|
| Kind of Alert | Serious, I recommend leaving the alert On with low priority |
| Root cause | Generated manually |
| Percentage | 11% about 52 alerts trigged |
| Whois Command | Amazon & Google who does have IIS Microsoft Server |



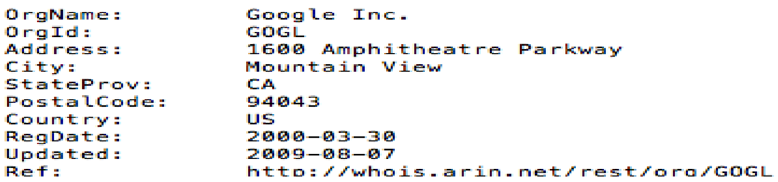
```
200 : 69 75 6B 59 69 51 49 61 73 62 45 32 69 52 5A 55   iukYiQIasbE2iRZU
210 : 41 41 41 41 42 26 61 64 5F 74 79 70 65 3D 69 66   AAAAB&ad_type=if
220 : 72 61 6D 65 26 61 64 5F 73 69 7A 65 3D 33 30 30   rame&ad_size=300
```



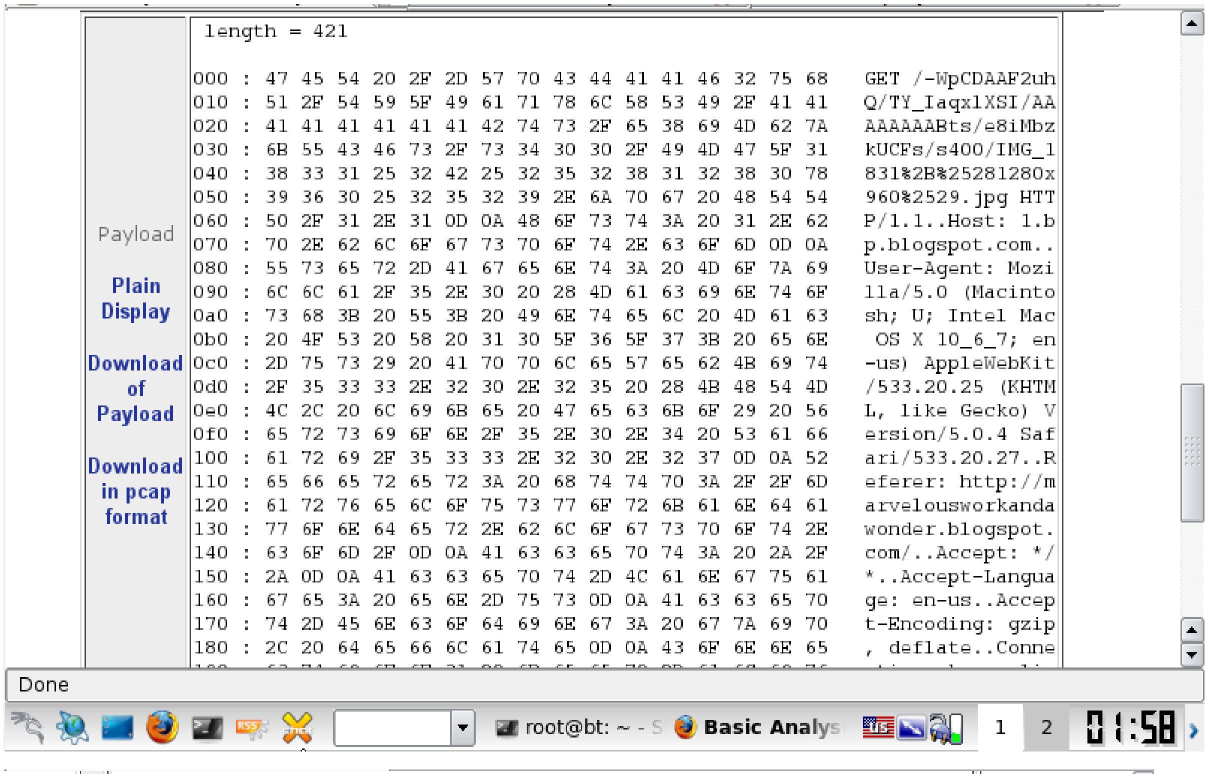
| | | | | |
|---|---|---|---|---|
| □ #34-(1-175)[snort] (http_inspect) DOUBLE DECODING ATTACK | 2011-04-11 00:04:40 | 192.168.5.247 | 174.129.128.117 | TCP |
| □ #35-(1-177)[snort] (http_inspect) DOUBLE DECODING ATTACK | 2011-04-11 00:04:40 | 192.168.5.247 | 50.17.147.248 | TCP |
| □ #36-(1-180)[snort] (http_inspect) DOUBLE DECODING ATTACK | 2011-04-11 00:04:42 | 192.168.5.72 | 64.94.107.12 | TCP |



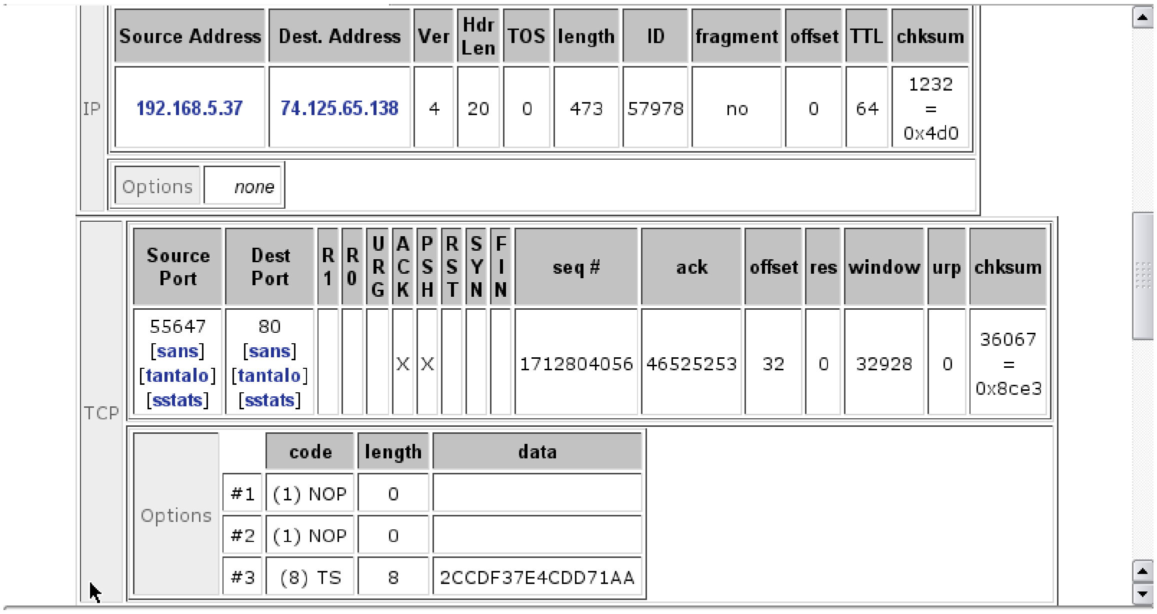
```
RNOCHandle: AN024-ARIN
RNOCName:   Amazon EC2 Network Operations
RNOCPhone:  +1-206-266-4064
RNOCEmail:  aes-noc@amazon.com
RNOCRef:    http://whois.arin.net/rest/poc/AN024-ARIN
```



```
OrgName:        Google Inc.
OrgId:          GOGL
Address:        1600 Amphitheatre Parkway
City:           Mountain View
StateProv:      CA
PostalCode:     94043
Country:        US
RegDate:        2000-03-30
Updated:        2009-08-07
Ref:            http://whois.arin.net/rest/org/GOGL
```



```
 length = 421

000 : 47 45 54 20 2F 2D 57 70 43 44 41 41 46 32 75 68   GET /-WpCDAAF2uh
010 : 51 2F 54 59 5F 49 61 71 78 6C 58 53 49 2F 41 41   Q/TY_IaqxlXSI/AA
020 : 41 41 41 41 41 41 42 74 73 2F 65 38 69 4D 62 7A   AAAAAABts/e8iMbz
030 : 6B 55 43 46 73 2F 73 34 30 30 2F 49 4D 47 5F 31   kUCFs/s400/IMG_1
040 : 38 33 31 25 32 42 25 32 35 32 38 31 32 38 30 78   831%2B%25281280x
050 : 39 36 30 25 32 35 32 39 2E 6A 70 67 20 48 54 54   960%2529.jpg HTT
060 : 50 2F 31 2E 31 0D 0A 48 6F 73 74 3A 20 31 2E 62   P/1.1..Host: 1.b
070 : 70 2E 62 6C 6F 67 73 70 6F 74 2E 63 6F 6D 0D 0A   p.blogspot.com..
080 : 55 73 65 72 2D 41 67 65 6E 74 3A 20 4D 6F 7A 69   User-Agent: Mozi
090 : 6C 6C 61 2F 35 2E 30 20 28 4D 61 63 69 6E 74 6F   lla/5.0 (Macinto
0a0 : 73 68 3B 20 55 3B 20 49 6E 74 65 6C 20 4D 61 63   sh; U; Intel Mac
0b0 : 20 4F 53 20 58 20 31 30 5F 36 5F 37 3B 20 65 6E    OS X 10_6_7; en
0c0 : 2D 75 73 29 20 41 70 70 6C 65 57 65 62 4B 69 74   -us) AppleWebKit
0d0 : 2F 35 33 33 2E 32 30 2E 32 35 20 28 4B 48 54 4D   /533.20.25 (KHTM
0e0 : 4C 2C 20 6C 69 6B 65 20 47 65 63 6B 6F 29 20 56   L, like Gecko) V
0f0 : 65 72 73 69 6F 6E 2F 35 2E 30 2E 34 20 53 61 66   ersion/5.0.4 Saf
100 : 61 72 69 2F 35 33 33 2E 32 30 2E 32 37 0D 0A 52   ari/533.20.27..R
110 : 65 66 65 72 65 72 3A 20 68 74 74 70 3A 2F 2F 6D   eferer: http://m
120 : 61 72 76 65 6C 6F 75 73 77 6F 72 6B 61 6E 64 61   arvelousworkanda
130 : 77 6F 6E 64 65 72 2E 62 6C 6F 67 73 70 6F 74 2E   wonder.blogspot.
140 : 63 6F 6D 2F 0D 0A 41 63 63 65 70 74 3A 20 2A 2F   com/..Accept: */
150 : 2A 0D 0A 41 63 63 65 70 74 2D 4C 61 6E 67 75 61   *..Accept-Langua
160 : 67 65 3A 20 65 6E 2D 75 73 0D 0A 41 63 63 65 70   ge: en-us..Accep
170 : 74 2D 45 6E 63 6F 64 69 6E 67 3A 20 67 7A 69 70   t-Encoding: gzip
180 : 2C 20 64 65 66 6C 61 74 65 0D 0A 43 6F 6E 6E 65   , deflate..Conne
```



| | Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 192.168.5.37 | 74.125.65.138 | 4 | 20 | 0 | 473 | 57978 | no | 0 | 64 | 1232 = 0x4d0 |

| Options | none |
|---|---|

| | Source Port | Dest Port | R 1 | R 0 | URG | ACK | PSH | RST | SYN | FIN | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TCP | 55647<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | X | | | | 1712804056 | 46525253 | 32 | 0 | 32928 | 0 | 36067 = 0x8ce3 |

| Options | | code | length | data |
|---|---|---|---|---|
| | #1 | (1) NOP | 0 | |
| | #2 | (1) NOP | 0 | |
| | #3 | (8) TS | 8 | 2CCDF37E4CDD71AA |

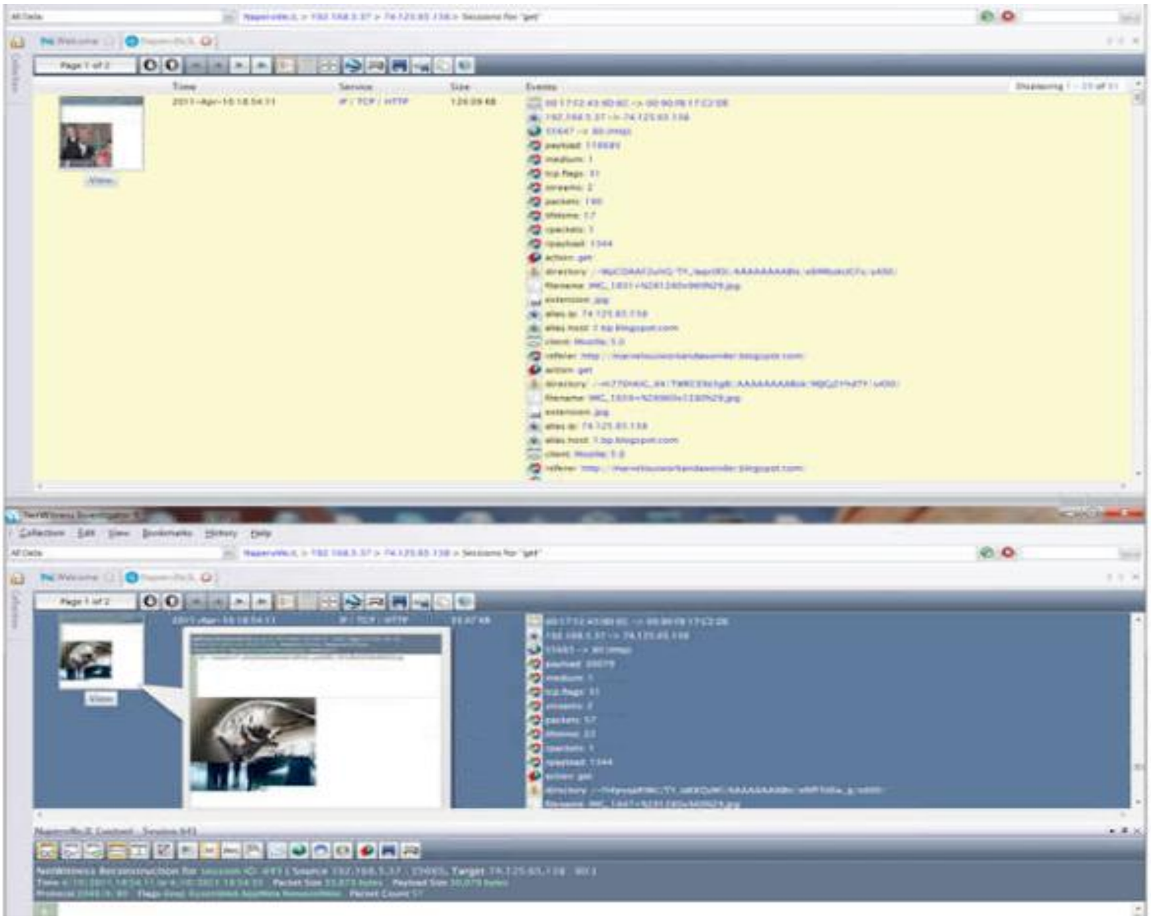
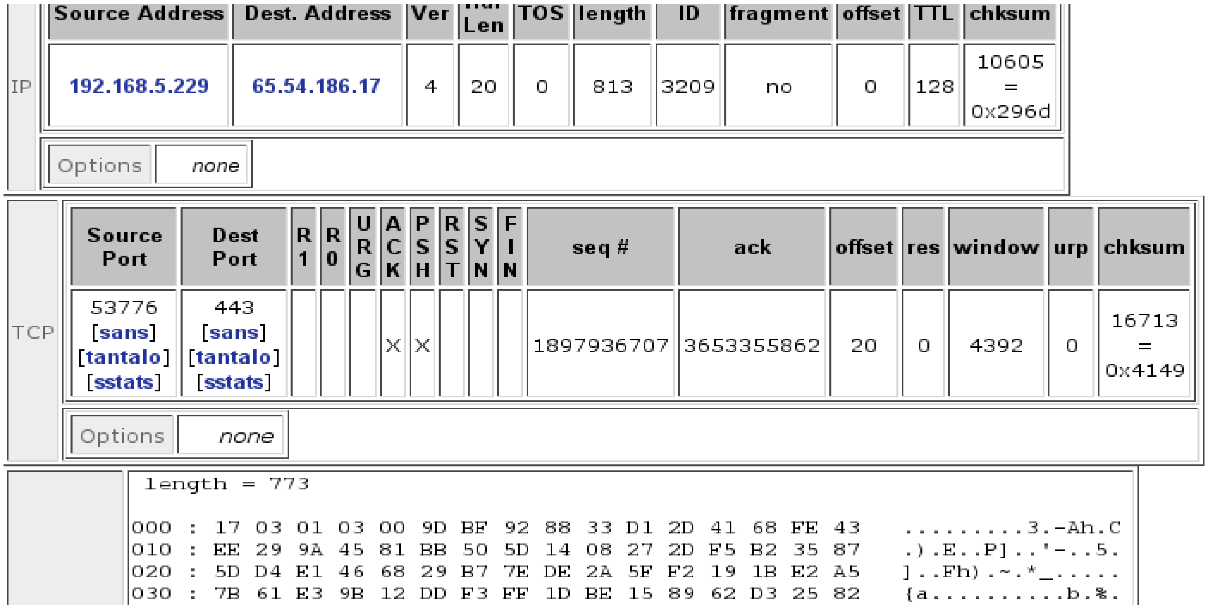


Figure 22:  HTTP Double Decoding Attack screenshots.

### 3- WEB-MISC SSLv2 Open SSL get shared ciphers overflow attempt

Often generated when an attempt is made to exploit a known vulnerability in an Open SSL implementation. Open SSL libraries are prone to a buffer overflow condition when processing user input. The SSL Get Shared Ciphers function reads data into a fixed length portion of memory; an attacker could utilize this vulnerability to execute code of their choosing on an affected system. Applications using the Open SSL libraries may also be prone to a Denial of Service Attack condition. Affected Systems are Open SSL libraries prior to 0.9.8d and Open SSL libraries prior to 0.9.7l. An attacker can supply excess data in the cipher exchange with a remote server to cause the overflow condition to be met. [28, 29]

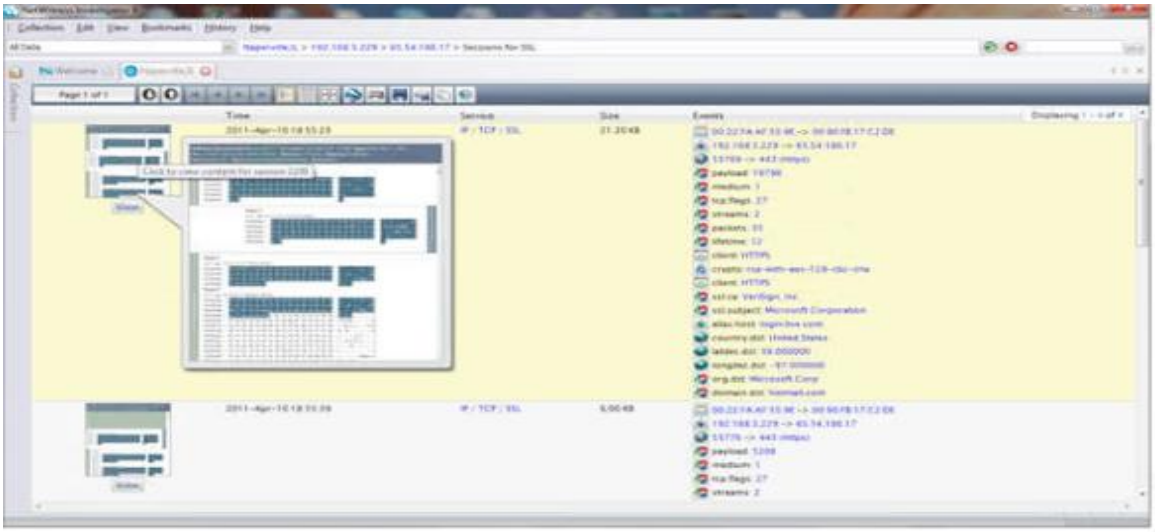
| WEB-MISC SSLv2 Open SSL get shared ciphers overflow | Analysis |
|---|---|
| Kind of Alert | Serious. Execution of this code is possible to cause Denial of Service attack (DOS). |
| Root cause | Automatically |
| Percentage | 19% 89 alerts went off |
| Whois Command | From internal IP address to MICROSOFT-1BLK server |



| IP | Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 192.168.5.229 | 65.54.186.17 | 4 | 20 | 0 | 813 | 3209 | no | 0 | 128 | 10605<br>=<br>0x296d |
| | Options | none | | | | | | | | | |

| TCP | Source Port | Dest Port | R1 | R0 | URG | ACK | PSH | RST | SYN | FIN | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 53776<br>[sans]<br>[tantalo]<br>[sstats] | 443<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | X | | | | 1897936707 | 3653355862 | 20 | 0 | 4392 | 0 | 16713<br>=<br>0x4149 |
| | Options | none | | | | | | | | | | | | | | | |

```
  length = 773

000 : 17 03 01 03 00 9D BF 92 88 33 D1 2D 41 68 FE 43   .........3.-Ah.C
010 : EE 29 9A 45 81 BB 50 5D 14 08 27 2D F5 B2 35 87   .).E..P]..'-..5.
020 : 5D D4 E1 46 68 29 B7 7E DE 2A 5F F2 19 1B E2 A5   ]..Fh).~.*_.....
030 : 7B 61 E3 9B 12 DD F3 FF 1D BE 15 89 62 D3 25 82   {a..........b.%.
```



```
[illegible] -> [illegible]
192.168.5.229 -> 65.54.186.17
53769 -> 443 (https)
payload: 19790
medium: 1
tcp.flags: 27
streams: 2
packets: 11
lifetime: 12
client: HTTPS
crypto: rsa-with-aes-128-cbc-sha
client: HTTPS
ssl.ca: VeriSign, Inc
ssl.subject: Microsoft Corporation
alias.host: login.live.com
country.dst: United States
latdec.dst: 38.000000
longdec.dst: -97.000000
org.dst: Microsoft Corp
domain.dst: hotmail.com

[illegible] -> [illegible]
192.168.5.229 -> 65.54.186.17
53776 -> 443 (https)
payload: 5208
medium: 1
tcp.flags: 27
streams: 2
```

Figure 23: WEB SSLv2 get shared ciphers overflow attempt screenshots.

**4- IIS Unicode CODEPOINT Encoding**

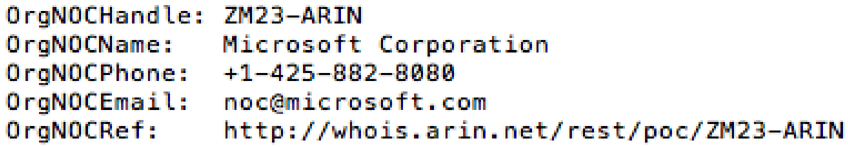
This event is generated when the pre-processor HTTP-Inspect detects Unicode encoded web requests. This may be an indicator of an obfuscated attack against a server as well as an attempt to evade an IDS. The Unicode map for the target servers can be generated for specific servers. Refer to the documentation for HTTP-Inspect for instructions. This event can be controlled using the HTTP_Inspect configuration options. [30]

| IIS Unicode CODEPOINT Encoding | Analysis |
|---|---|
| Kind of Alert | Unknown, but I think we should leave it on with mid priority tag. |
| Root cause | Generated manually |
| Percentage | 0% 2 alerts went off |
| Whois Command | Both addresses “Source & Destination” are private |

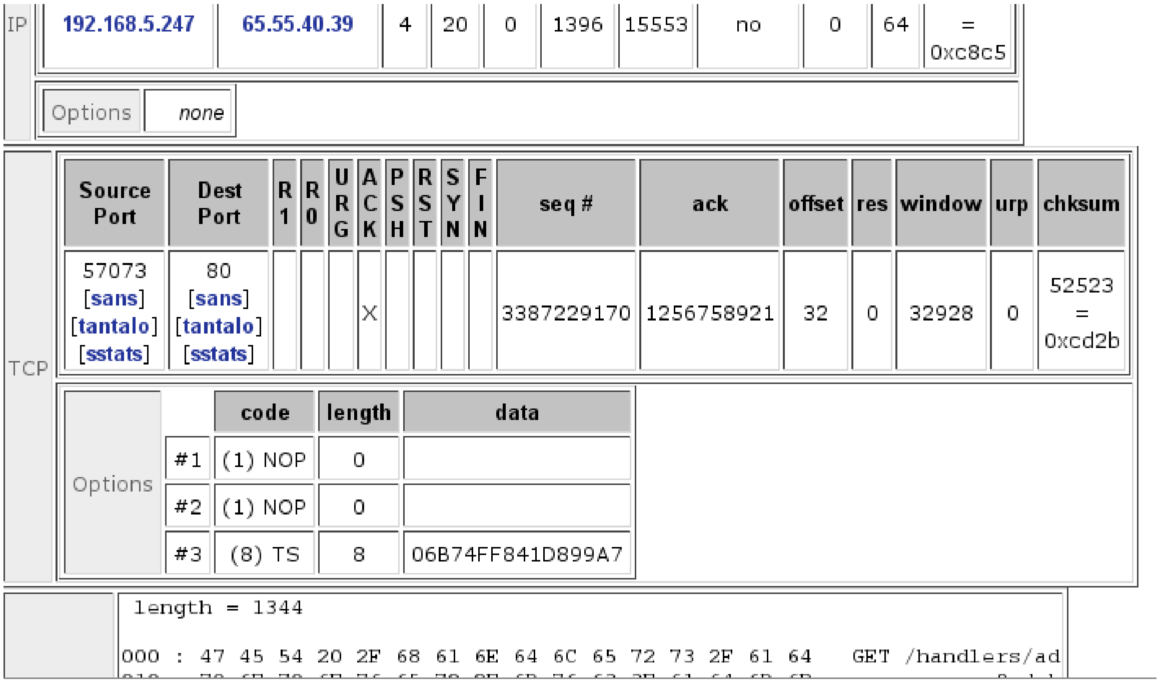
**5- WEB-MISC handler access**

This event, Web Application Misalliance, is generated when an attempt is made to exploit a known vulnerability on a web server or a web application. Some applications do not perform stringent checks when validating the credentials of a client host connecting to the services offered on a host server. This can lead to unauthorized access and possibly escalated privileges to the administrator. Data stored on the machine can be compromised and the attacker can exploit trust relationships between the victim server and other hosts. It causes information gathering, system integrity compromise, possible unauthorized administrative access to the server and possible execution of arbitrary code of the attackers choosing in some cases. As a corrective action, ensure the system is using an up-to-date version of the software and has had all vendor supplied patches applied. Check the host-log files and application logs for signs of compromise or abnormal behaviors. [31]

| WEB-MISC handler access | Analysis |
|---|---|
| Kind of Alert | False positive since I see my IP address (192.168.5.247) generated the alert as the screenshots show. If Starbucks web server has pages by the name of calendar* this rule will fire often. Many sites now use calendar applications and this rule may generate a large number of false positives alert. So I'd recommend avoid the confusing, and be more specific when writing the rules. |
| Root cause | Automatically generated |
| Percentage | 2% 10 alerts went off |
| Whois Command | Microsoft Corporation |



```
OrgNOCHandle:  ZM23-ARIN
OrgNOCName:    Microsoft Corporation
OrgNOCPhone:   +1-425-882-8080
OrgNOCEmail:   noc@microsoft.com
OrgNOCRef:     http://whois.arin.net/rest/poc/ZM23-ARIN
```



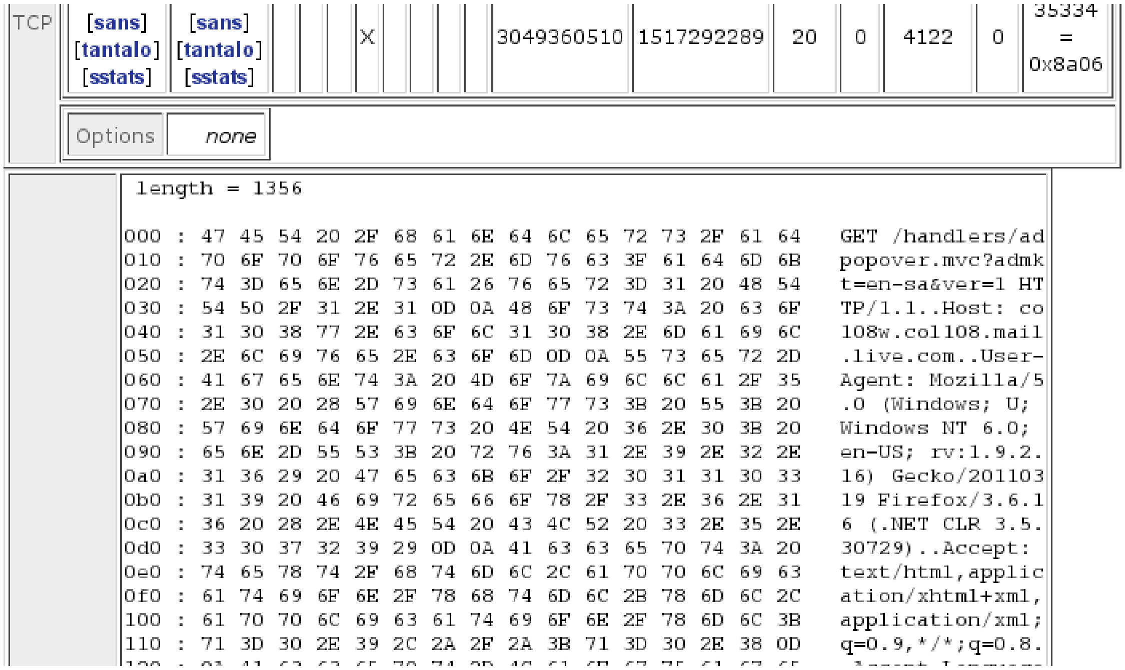
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 192.168.5.247 | 65.55.40.39 | 4 | 20 | 0 | 1396 | 15553 | no | 0 | 64 | =<br>0xc8c5 |

| | |
|---|---|
| Options | *none* |

| | Source Port | Dest Port | R 1 | R 0 | U R G | A C K | P S H | R S T | S Y N | F I N | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TCP | 57073<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | | | | | 3387229170 | 1256758921 | 32 | 0 | 32928 | 0 | 52523<br>=<br>0xcd2b |

| | | code | length | data |
|---|---|---|---|---|
| Options | #1 | (1) NOP | 0 | |
| | #2 | (1) NOP | 0 | |
| | #3 | (8) TS | 8 | 06B74FF841D899A7 |

```
 length = 1344

000 : 47 45 54 20 2F 68 61 6E 64 6C 65 72 73 2F 61 64   GET /handlers/ad
```



| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TCP | [sans]<br>[tantalo]<br>[sstats] | [sans]<br>[tantalo]<br>[sstats] | | | X | | | | | 3049360510 | 1517292289 | 20 | 0 | 4122 | 0 | 35334<br>=<br>0x8a06 |
| | Options | none | | | | | | | | | | | | | | |

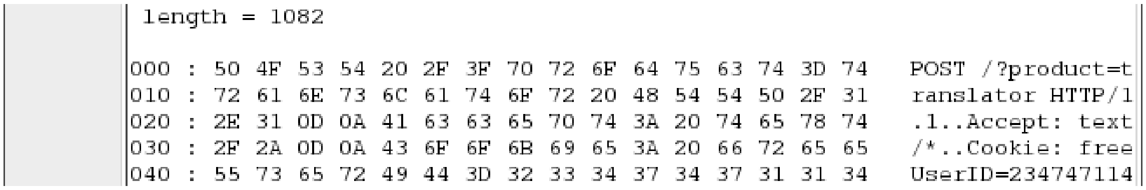
```
length = 1356

000 : 47 45 54 20 2F 68 61 6E 64 6C 65 72 73 2F 61 64   GET /handlers/ad
010 : 70 6F 70 6F 76 65 72 2E 6D 76 63 3F 61 64 6D 6B   popover.mvc?admk
020 : 74 3D 65 6E 2D 73 61 26 76 65 72 3D 31 20 48 54   t=en-sa&ver=1 HT
030 : 54 50 2F 31 2E 31 0D 0A 48 6F 73 74 3A 20 63 6F   TP/1.1..Host: co
040 : 31 30 38 77 2E 63 6F 6C 31 30 38 2E 6D 61 69 6C   108w.col108.mail
050 : 2E 6C 69 76 65 2E 63 6F 6D 0D 0A 55 73 65 72 2D   .live.com..User-
060 : 41 67 65 6E 74 3A 20 4D 6F 7A 69 6C 6C 61 2F 35   Agent: Mozilla/5
070 : 2E 30 20 28 57 69 6E 64 6F 77 73 3B 20 55 3B 20   .0 (Windows; U;
080 : 57 69 6E 64 6F 77 73 20 4E 54 20 36 2E 30 3B 20   Windows NT 6.0;
090 : 65 6E 2D 55 53 3B 20 72 76 3A 31 2E 39 2E 32 2E   en-US; rv:1.9.2.
0a0 : 31 36 29 20 47 65 63 6B 6F 2F 32 30 31 31 30 33   16) Gecko/201103
0b0 : 31 39 20 46 69 72 65 66 6F 78 2F 33 2E 36 2E 31   19 Firefox/3.6.1
0c0 : 36 20 28 2E 4E 45 54 20 43 4C 52 20 33 2E 35 2E   6 (.NET CLR 3.5.
0d0 : 33 30 37 32 39 29 0D 0A 41 63 63 65 70 74 3A 20   30729)..Accept:
0e0 : 74 65 78 74 2F 68 74 6D 6C 2C 61 70 70 6C 69 63   text/html,applic
0f0 : 61 74 69 6F 6E 2F 78 68 74 6D 6C 2B 78 6D 6C 2C   ation/xhtml+xml,
100 : 61 70 70 6C 69 63 61 74 69 6F 6E 2F 78 6D 6C 3B   application/xml;
110 : 71 3D 30 2E 39 2C 2A 2F 2A 3B 71 3D 30 2E 38 0D   q=0.9,*/*;q=0.8.
```

Figure 24: WEB-MISC handler access screenshots.

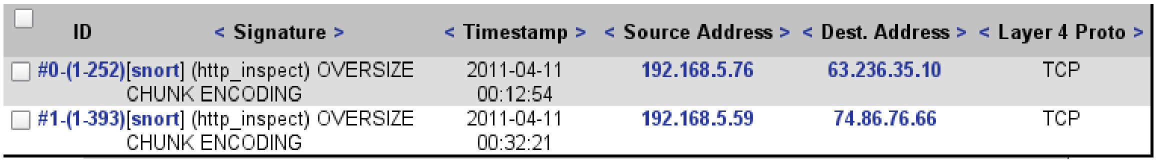
### 6- Oversize Chunk Encoding

This event is generated when the pre-processor HTTP-Inspect detects network traffic that may constitute an attack. In particular, this attack is generated when the HTTP-Inspect detects the use of an oversized chunk encoded request. This may be an indicator of an attack against a web server. This event may also indicate the use of HTTP tunneling. This event can be controlled using the HTTP Inspect configuration properly. [32]

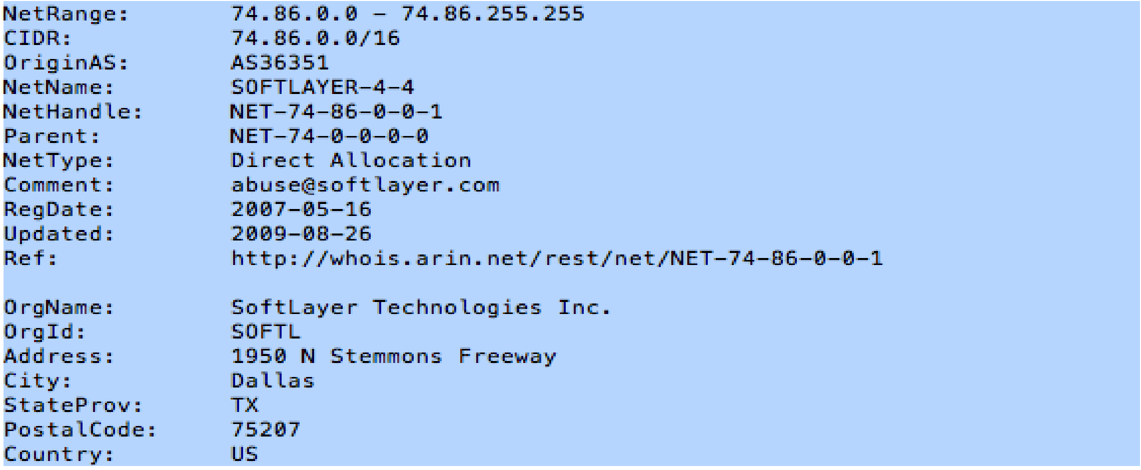
| Oversize Chunk Encoding | Analysis |
|---|---|
| Kind of Alert | True Positive. No browser makes malicious requests, codes, or size. And I realized that both requests have the same server feedback "Post /?product=translator HTTP/1.1.." It seems that someone with this IP address of (192.168.5.76) is using an oversized chunk encoded request to both destination. Starbucks might held responsible in case where Microsoft or Softlyer server got hacked |
| Root cause | Generated manually |
| Percentage | 0% 2 alerts went off |
| Whois Command | Internal IP requested HTTP get to both Qwest & SoftLayer technology Co. By looking to Netwitness tool (192.168.5.76) was actually going to Fox News |



```
length = 1082

000 : 50 4F 53 54 20 2F 3F 70 72 6F 64 75 63 74 3D 74   POST /?product=t
010 : 72 61 6E 73 6C 61 74 6F 72 20 48 54 54 50 2F 31   ranslator HTTP/1
020 : 2E 31 0D 0A 41 63 63 65 70 74 3A 20 74 65 78 74   .1..Accept: text
030 : 2F 2A 0D 0A 43 6F 6F 6B 69 65 3A 20 66 72 65 65   /*..Cookie: free
040 : 55 73 65 72 49 44 3D 32 33 34 37 34 37 31 31 34   UserID=234747114
```



| □ ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|
| □ #0-(1-252) | [snort] (http_inspect) OVERSIZE CHUNK ENCODING | 2011-04-11 00:12:54 | 192.168.5.76 | 63.236.35.10 | TCP |
| □ #1-(1-393) | [snort] (http_inspect) OVERSIZE CHUNK ENCODING | 2011-04-11 00:32:21 | 192.168.5.59 | 74.86.76.66 | TCP |



```
NetRange:       74.86.0.0 - 74.86.255.255
CIDR:           74.86.0.0/16
OriginAS:       AS36351
NetName:        SOFTLAYER-4-4
NetHandle:      NET-74-86-0-0-1
Parent:         NET-74-0-0-0-0
NetType:        Direct Allocation
Comment:        abuse@softlayer.com
RegDate:        2007-05-16
Updated:        2009-08-26
Ref:            http://whois.arin.net/rest/net/NET-74-86-0-0-1

OrgName:        SoftLayer Technologies Inc.
OrgId:          SOFTL
Address:        1950 N Stemmons Freeway
City:           Dallas
StateProv:      TX
PostalCode:     75207
Country:        US
```

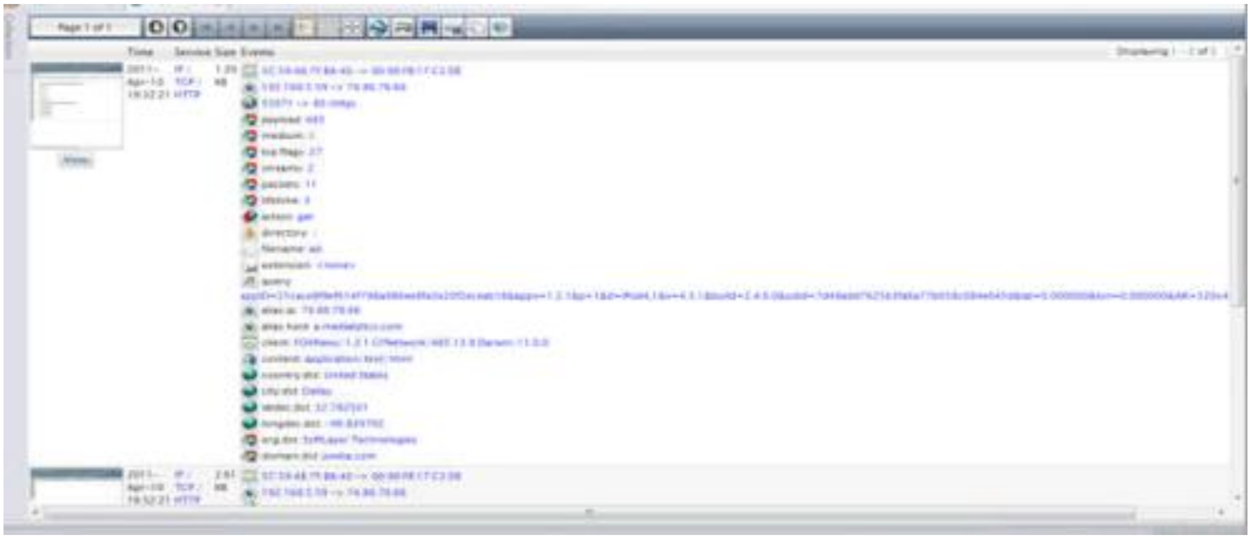
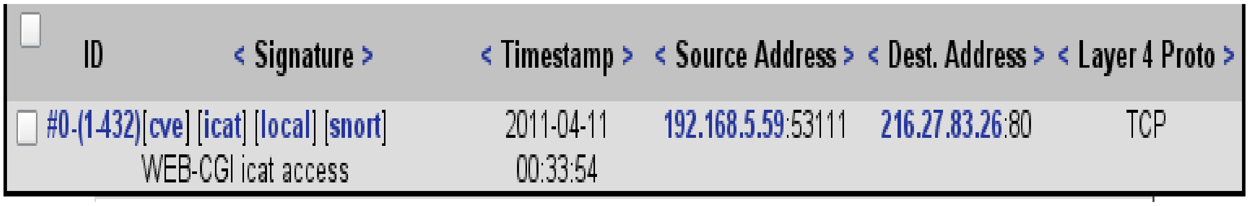


Figure 25: Oversize Chunk Encoding screenshots.

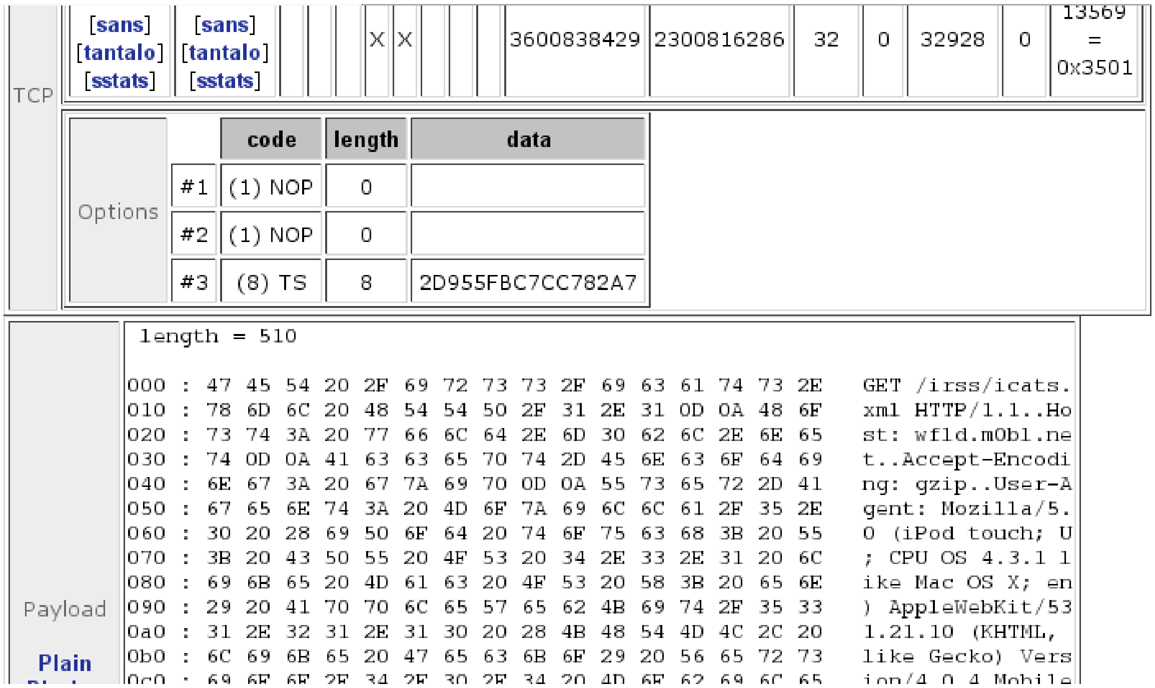
### 7- WEB-CGI icat access with 0%

This is a known vulnerability in a CGI web application running on a server. This event is generated when an attempt is made to exploit and gain unauthorized access to a CGI application running on a web server. There is no stringent check process to validate the credentials of clients connecting to the CGI web applications hosted by a server. Impact can lead to unauthorized access and possibly escalated privileges to that of the administrator. Data stored on the machine can be compromised and the attacker can exploit trust relationships between the victim server and other hosts. If stringent input checks are not performed by the CGI application, it may also be possible for an attacker to execute system binaries or malicious code of the attackers choosing. As an attacker can access an authentication mechanism and supply his/her own credentials to gain access. Alternatively the attacker can exploit weaknesses to gain access as the administrator by supplying input of their choosing to the underlying CGI script. [33]

| WEB-CGI icat access | Analysis |
|---|---|
| Kind of Alert | Serious. Administrator should silently drop that request, and block that particular source IP address. |
| Root cause | Generated manually |
| Percentage | 0% one alert |
| Whois Command | Destination is Hosted Solutions Acquisition, LLC that is running CGi application, as you can see the second screenshot that has the Get /irss/icats. Xml code. |



| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-432) | [cve] [icat] [local] [snort] WEB-CGI icat access | 2011-04-11 00:33:54 | 192.168.5.59:53111 | 216.27.83.26:80 | TCP |



TCP

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| [sans]<br>[tantalo]<br>[sstats] | [sans]<br>[tantalo]<br>[sstats] | | | | X | X | | | | 3600838429 | 2300816286 | 32 | 0 | 32928 | 0 | 13569<br>=<br>0x3501 |

Options

| | code | length | data |
|---|---|---|---|
| #1 | (1) NOP | 0 | |
| #2 | (1) NOP | 0 | |
| #3 | (8) TS | 8 | 2D955FBC7CC782A7 |

Payload

Plain

```
 length = 510

000 : 47 45 54 20 2F 69 72 73 73 2F 69 63 61 74 73 2E   GET /irss/icats.
010 : 78 6D 6C 20 48 54 54 50 2F 31 2E 31 0D 0A 48 6F   xml HTTP/1.1..Ho
020 : 73 74 3A 20 77 66 6C 64 2E 6D 30 62 6C 2E 6E 65   st: wfld.m0bl.ne
030 : 74 0D 0A 41 63 63 65 70 74 2D 45 6E 63 6F 64 69   t..Accept-Encodi
040 : 6E 67 3A 20 67 7A 69 70 0D 0A 55 73 65 72 2D 41   ng: gzip..User-A
050 : 67 65 6E 74 3A 20 4D 6F 7A 69 6C 6C 61 2F 35 2E   gent: Mozilla/5.
060 : 30 20 28 69 50 6F 64 20 74 6F 75 63 68 3B 20 55   0 (iPod touch; U
070 : 3B 20 43 50 55 20 4F 53 20 34 2E 33 2E 31 20 6C   ; CPU OS 4.3.1 l
080 : 69 6B 65 20 4D 61 63 20 4F 53 20 58 3B 20 65 6E   ike Mac OS X; en
090 : 29 20 41 70 70 6C 65 57 65 62 4B 69 74 2F 35 33   ) AppleWebKit/53
0a0 : 31 2E 32 31 2E 31 30 20 28 4B 48 54 4D 4C 2C 20   1.21.10 (KHTML, 
0b0 : 6C 69 6B 65 20 47 65 63 6B 6F 29 20 56 65 72 73   like Gecko) Vers
0c0 : 69 6F 6E 2F 34 2E 30 2E 34 20 4D 6F 62 69 6C 65   ion/4.0.4 Mobile
```

Figure 26: Web CGI Access screenshots.

## Recommendations of Web Cyber Threat

Figure 27 summarizes the recommended actions for each of the threats described in this chapter.

| Web Threats | Recommended Action |
|---|---|
| **HTTP-Inspect Double Decoding Attack** | Check the target host for signs of compromise. Apply any appropriate vendor supplied patches. Upgrade to the latest non-affected version of the software Use Apache. In addition, reconfiguring HTTP inspector for proper filtering function |
| **WEB-CGI calendar access**<br>**OR**<br>**WEB-CGI icat access** | Check the target host for signs of compromise. Ensure the system is using an up to date version of the software and has had all vendor supplied patches applied. If your web server has pages by the name of calendar* this rule will fire often. Probably, Starbuck's server use calendar applications and this rule may generate a large number of false positives, it does not distinguish between perl cgi applications and php scripts because of purely written rules that need to be tuned. Consider tuning this rule for your site, and changing the name of the script to something other than "calendar". |
| **Open SSL get shared ciphers overflow attempt** | Upgrade to the latest non-affected Open SSL libraries and recompile any software that uses the libraries. |
| **IIS Unicode CODEPOINT Encoding** | Check the target host for signs of compromise. Apply any appropriate vendor supplied patches. |
| **WEB-MISC handler access** | Ensure the system is using an up to date version of the software and has had all vendor supplied patches applied. Check the host log files and application logs for any sign of compromise. |
| **Oversize Chunk Encoding** | We have to make sure if there is any event trigged on each host. Whenever we have this code "Post /?product=translator HTTP/1.1." we have to configure HTTP Inspect properly. In case, it's just a noise alert, we should tune it by rewriting the rules. |

Figure 27: Recommendations of Web Cyber Threats.

# Chapter 6: Server Cyber Threats:

This section discusses server cyber threats and provides background information on server security. It covers the following server threats that are found on Public hotspot wifi:

- Open Port Scan
- Bare Byte Unicode Decoding
- HTTP Inspect Oversize Request
- TCP Port sweep
- HTTP-Inspect U Encoding
- WEB SSLv3 invalid data version attempt
- MISC IBM Lotus Domino WEB Server Accept-Language header buffer Overflow

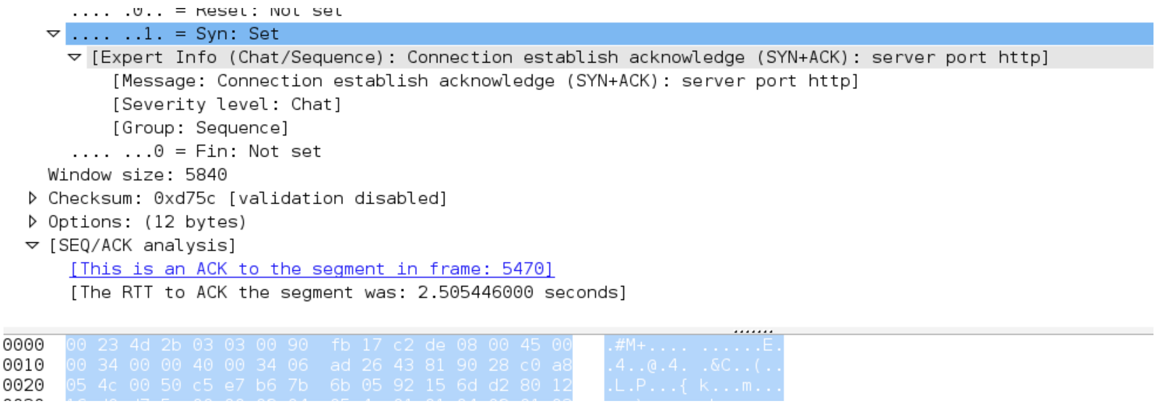
The following will be discussed for each threat:

- The type of the threat
- Description of the threats
- The possible threat scenario
- The type of the alert
- The root cause of the incident
- The representative percentages
- If possible who initiated that threat and its recipients
- The recommended Action for administrator

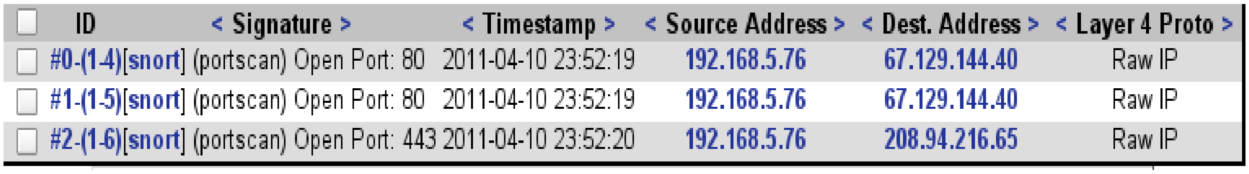
### 1- Open Port Scan

Open Port scan alert is generated in a point where the pre-processor sfPortscan detects network traffic that may constitute an attack. A sfportscan is a pre-processor that detects network traffic which may constitute an attack; specifically, an open port was detected. This is normally an indicator of possible network reconnaissance and may be the prelude to a targeted attack against the targeted system. A port scan is often the first stage in a targeted attack against a system. An attacker can use different port scanning techniques and tools to determine the target host operating system and application versions running on the host to determine the possible attack vectors against that host. In particular, a hacker often uses a port scanning technique, which is illegal in the United States, to determine operating system type and version. Also application versions can be identified to determine possible effective attack vectors that can be used against the target host. In this case the scanner was able to get the server type, version, and the running application type as shown below. [34]

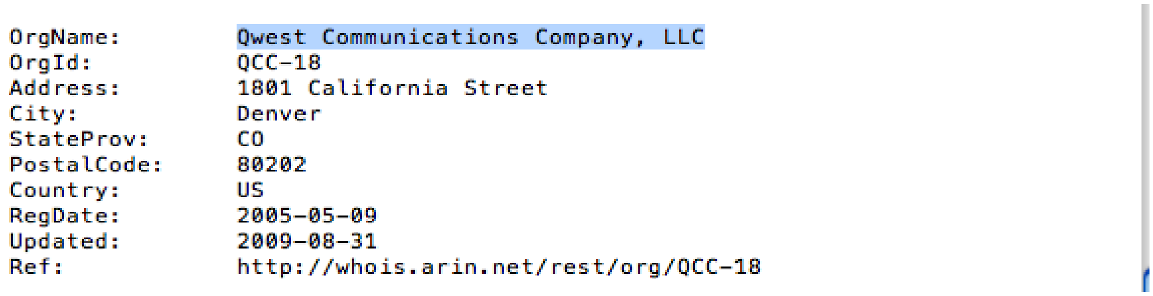
| Open Port Scan | Analysis |
|---|---|
| Kind of Alert | True Positive. |
| Root cause | Generated automatically with IP address of (192.169.5.76) scanned for open ports on two different hosts on ports 80 & 443 |
| Percentage | 1% about 3 alerts trigged |
| Whois Command | Source is internal, targets was Qwest Carrier & Rearden Commerce |



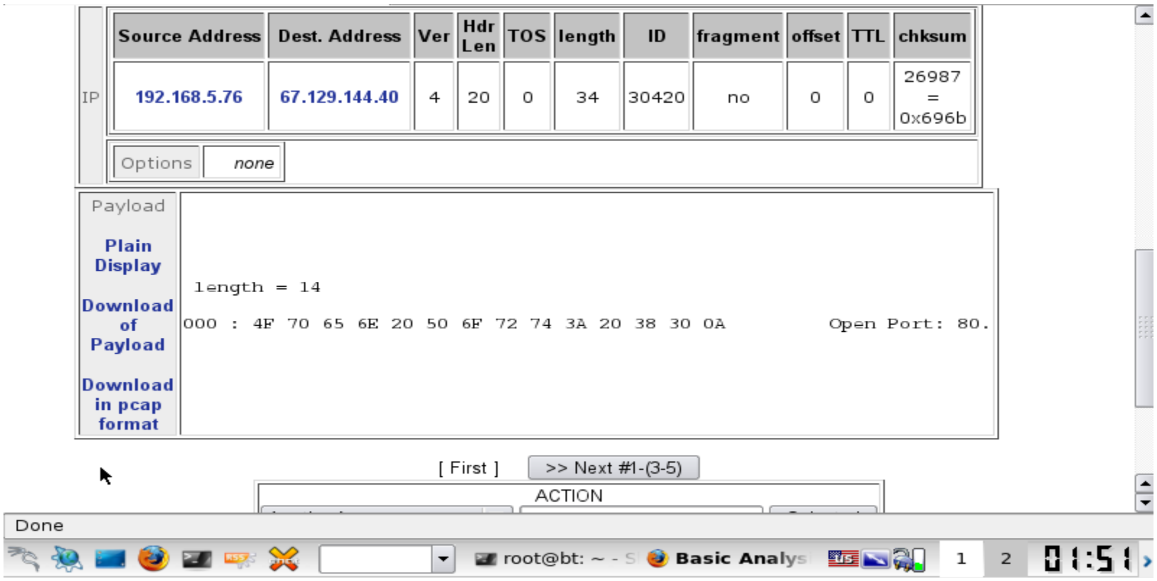
```
    .... .0.. = Reset: Not set
  ▽ .... ..1. = Syn: Set
    ▽ [Expert Info (Chat/Sequence): Connection establish acknowledge (SYN+ACK): server port http]
        [Message: Connection establish acknowledge (SYN+ACK): server port http]
        [Severity level: Chat]
        [Group: Sequence]
    .... ...0 = Fin: Not set
  Window size: 5840
▷ Checksum: 0xd75c [validation disabled]
▷ Options: (12 bytes)
▽ [SEQ/ACK analysis]
    [This is an ACK to the segment in frame: 5470]
    [The RTT to ACK the segment was: 2.505446000 seconds]

0000  00 23 4d 2b 03 03 00 90  fb 17 c2 de 08 00 45 00   .#M+.... ......E.
0010  00 34 00 00 40 00 34 06  ad 26 43 81 90 28 c0 a8   .4..@.4. .&C..(..
0020  05 4c 00 50 c5 e7 b6 7b  6b 05 92 15 6d d2 80 12   .L.P...{ k...m...
```



| □ ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|
| □ #0-(1-4)[snort] | (portscan) Open Port: 80 | 2011-04-10 23:52:19 | 192.168.5.76 | 67.129.144.40 | Raw IP |
| □ #1-(1-5)[snort] | (portscan) Open Port: 80 | 2011-04-10 23:52:19 | 192.168.5.76 | 67.129.144.40 | Raw IP |
| □ #2-(1-6)[snort] | (portscan) Open Port: 443 | 2011-04-10 23:52:20 | 192.168.5.76 | 208.94.216.65 | Raw IP |



```
OrgName:        Qwest Communications Company, LLC
OrgId:          QCC-18
Address:        1801 California Street
City:           Denver
StateProv:      CO
PostalCode:     80202
Country:        US
RegDate:        2005-05-09
Updated:        2009-08-31
Ref:            http://whois.arin.net/rest/org/QCC-18
```



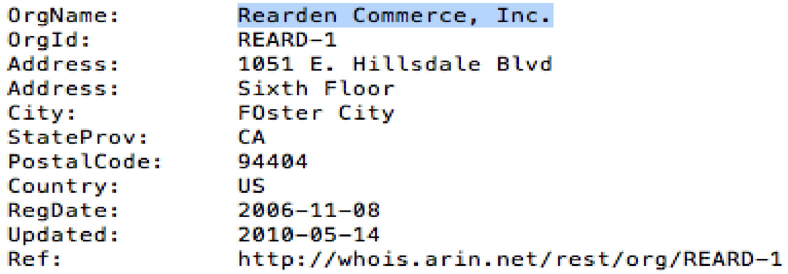
IP

| Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|
| 192.168.5.76 | 67.129.144.40 | 4 | 20 | 0 | 34 | 30420 | no | 0 | 0 | 26987<br>=<br>0x696b |

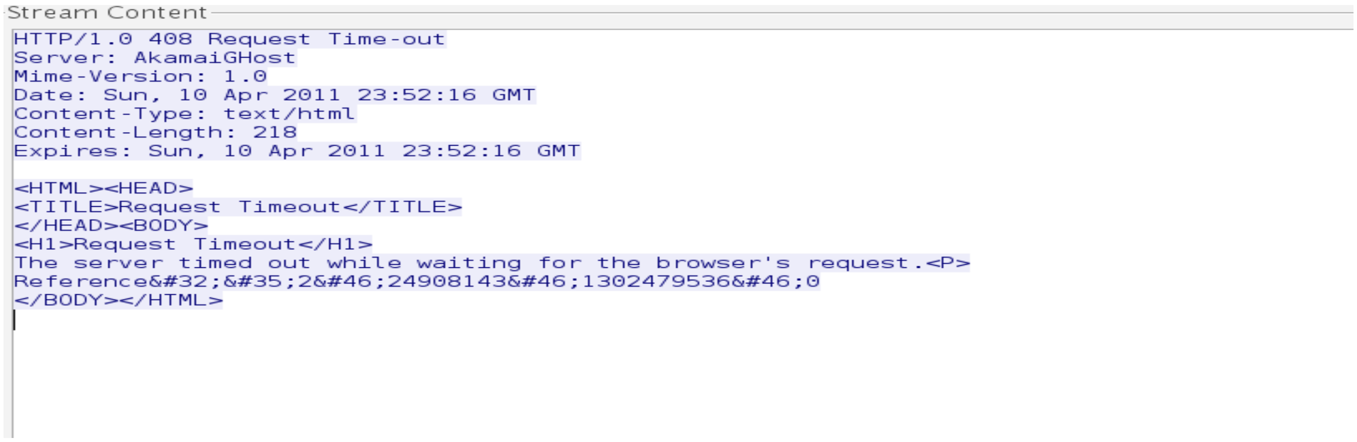
| Options | none |
|---|---|

Payload

```
 length = 14
000 : 4F 70 65 6E 20 50 6F 72 74 3A 20 38 30 0A          Open Port: 80.
```



```
OrgName:        Rearden Commerce, Inc.
OrgId:          REARD-1
Address:        1051 E. Hillsdale Blvd
Address:        Sixth Floor
City:           FOster City
StateProv:      CA
PostalCode:     94404
Country:        US
RegDate:        2006-11-08
Updated:        2010-05-14
Ref:            http://whois.arin.net/rest/org/REARD-1
```



Stream Content

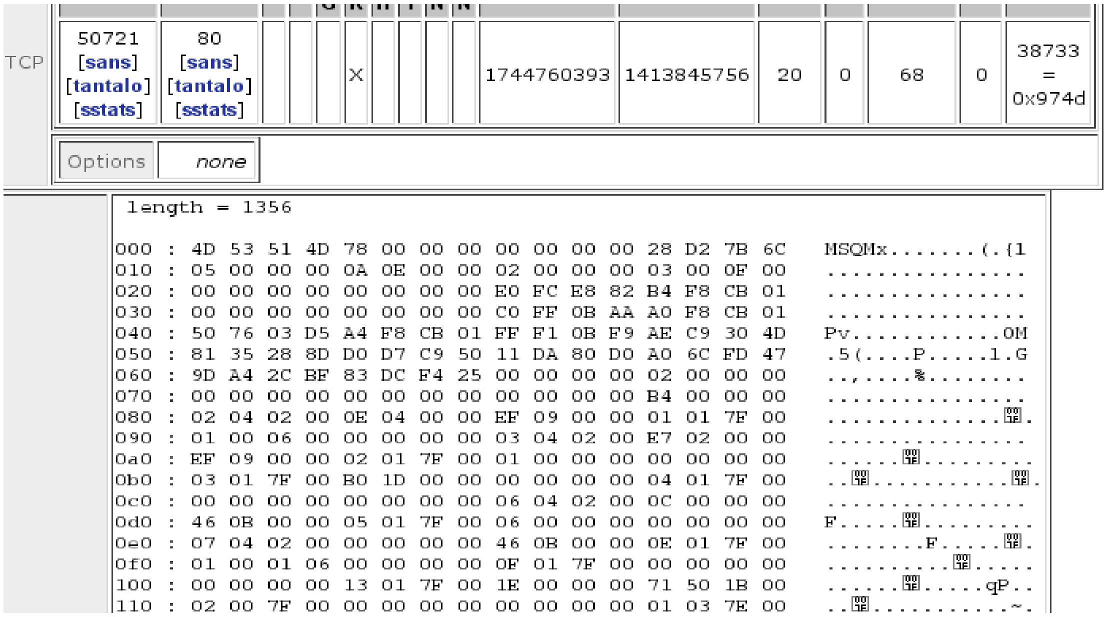
```
HTTP/1.0 408 Request Time-out
Server: AkamaiGHost
Mime-Version: 1.0
Date: Sun, 10 Apr 2011 23:52:16 GMT
Content-Type: text/html
Content-Length: 218
Expires: Sun, 10 Apr 2011 23:52:16 GMT

<HTML><HEAD>
<TITLE>Request Timeout</TITLE>
</HEAD><BODY>
<H1>Request Timeout</H1>
The server timed out while waiting for the browser's request.<P>
Reference&#32;&#35;2&#46;24908143&#46;1302479536&#46;0
</BODY></HTML>
```

Figure 28: Open Port Scan Threat Screenshots.

### 2- Bare Byte Unicode Decoding

Microsoft IIS servers are able to use non-ASCII characters as values when decoding UTF-8 values. This is non-standard behavior for a Web Server and violates RFC recommendations. All non-ASCII values should be encoded with a %. This event may indicate an attack against a web server or at the least an attempt to evade Intrusion Detection System, since no web clients encode UTF-8 characters this way, which is likely a malicious request. This event can be controlled using proper HTTP-INSPECT configurations. The only way an attacker can lunch a successful attack is by encoding a web request using this non-standard format to perform. [35]

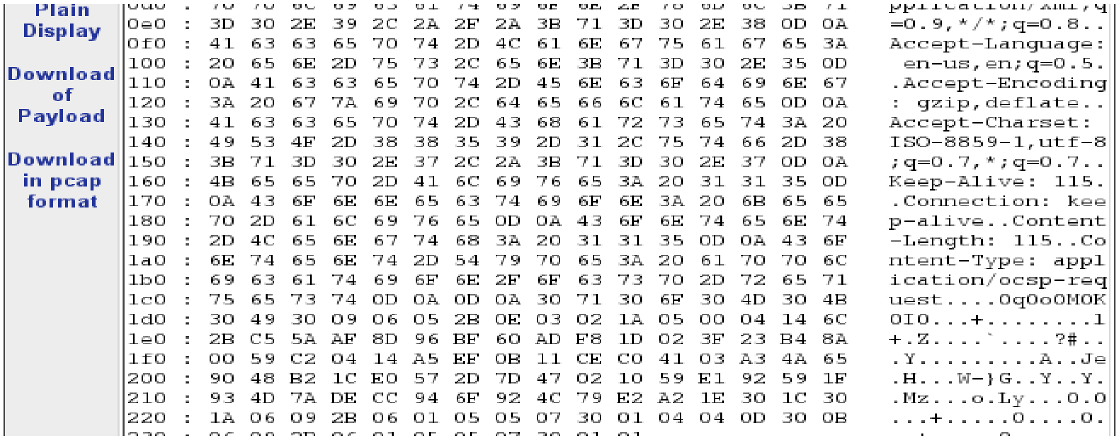
| Bare Byte Unicode Decoding | Analysis |
|---|---|
| Kind of Alert | True Positive |
| Root cause | Generated manually |
| Percentage | 1% 7 alerts trigged |
| Whois Command | Both addresses “Source & Destination” are private |



| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TCP | 50721<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | | | | | 1744760393 | 1413845756 | 20 | 0 | 68 | 0 | 38733<br>=<br>0x974d |

| | |
|---|---|
| Options | *none* |

```
 length = 1356

000 : 4D 53 51 4D 78 00 00 00 00 00 00 00 28 D2 7B 6C   MSQMx.......(.{l
010 : 05 00 00 00 0A 0E 00 00 02 00 00 00 03 00 0F 00   ................
020 : 00 00 00 00 00 00 00 00 E0 FC E8 82 B4 F8 CB 01   ................
030 : 00 00 00 00 00 00 00 00 C0 FF 0B AA A0 F8 CB 01   ................
040 : 50 76 03 D5 A4 F8 CB 01 FF F1 0B F9 AE C9 30 4D   Pv............0M
050 : 81 35 28 8D D0 D7 C9 50 11 DA 80 D0 A0 6C FD 47   .5(....P.....l.G
060 : 9D A4 2C BF 83 DC F4 25 00 00 00 00 02 00 00 00   ..,....%........
070 : 00 00 00 00 00 00 00 00 00 00 00 00 B4 00 00 00   ................
080 : 02 04 02 00 0E 04 00 00 EF 09 00 00 01 01 7F 00   ..............□.
090 : 01 00 06 00 00 00 00 00 03 04 02 00 E7 02 00 00   ................
0a0 : EF 09 00 00 02 01 7F 00 01 00 00 00 00 00 00 00   ......□.........
0b0 : 03 01 7F 00 B0 1D 00 00 00 00 00 00 04 01 7F 00   ..□...........□.
0c0 : 00 00 00 00 00 00 00 00 06 04 02 00 0C 00 00 00   ................
0d0 : 46 0B 00 00 05 01 7F 00 06 00 00 00 00 00 00 00   F.....□.........
0e0 : 07 04 02 00 00 00 00 00 46 0B 00 00 0E 01 7F 00   ........F.....□.
0f0 : 01 00 01 06 00 00 00 00 0F 01 7F 00 00 00 00 00   ..........□.....
100 : 00 00 00 00 13 01 7F 00 1E 00 00 00 71 50 1B 00   ......□.....qP..
110 : 02 00 7F 00 00 00 00 00 00 00 00 00 01 03 7E 00   ..□...........~.
```



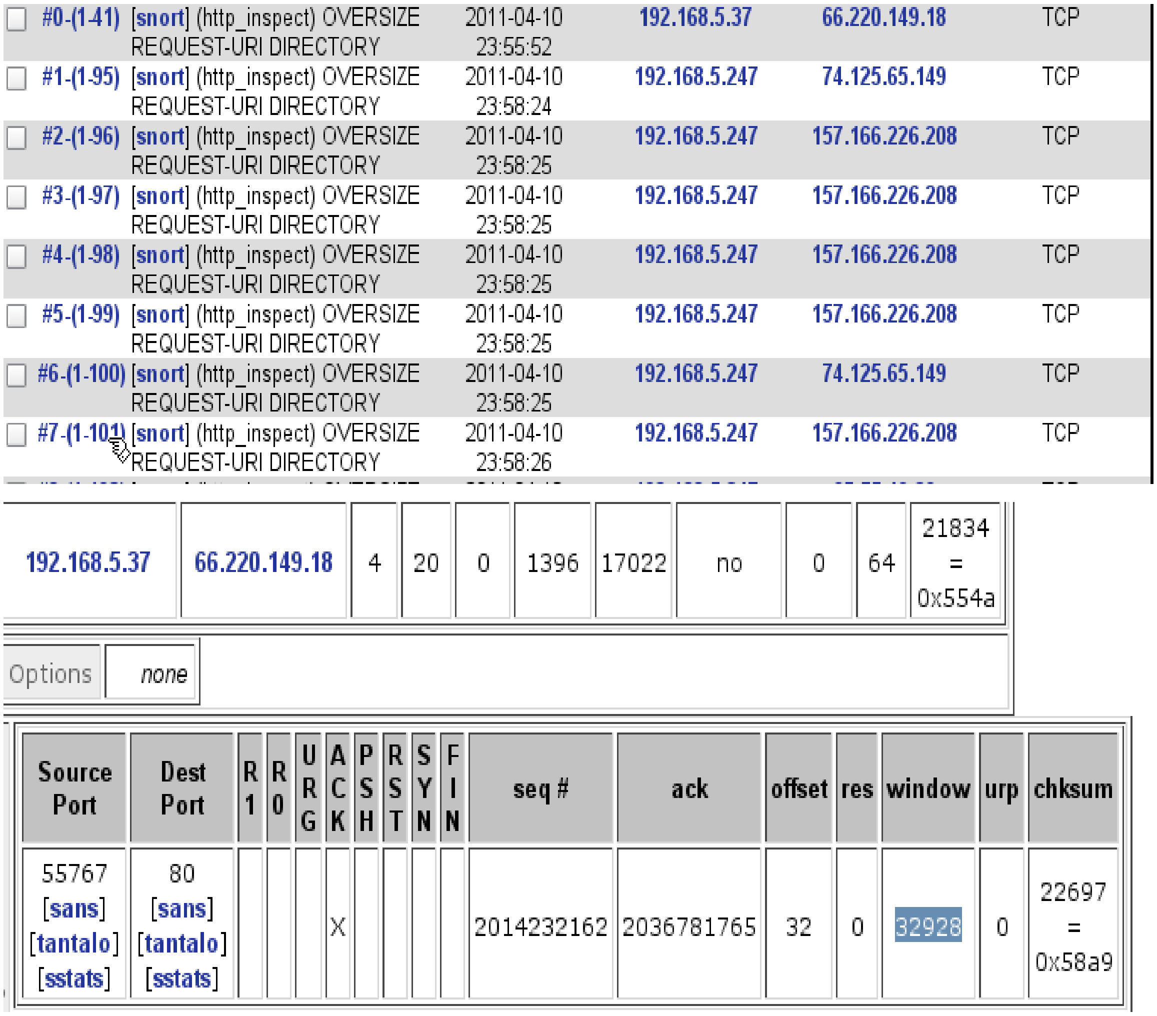
```
0d0 : 70 70 6C 69 63 61 74 69 6F 6E 2F 78 6D 6C 3B 71   pplication/xml;q
0e0 : 3D 30 2E 39 2C 2A 2F 2A 3B 71 3D 30 2E 38 0D 0A   =0.9,*/*;q=0.8..
0f0 : 41 63 63 65 70 74 2D 4C 61 6E 67 75 61 67 65 3A   Accept-Language:
100 : 20 65 6E 2D 75 73 2C 65 6E 3B 71 3D 30 2E 35 0D    en-us,en;q=0.5.
110 : 0A 41 63 63 65 70 74 2D 45 6E 63 6F 64 69 6E 67   .Accept-Encoding
120 : 3A 20 67 7A 69 70 2C 64 65 66 6C 61 74 65 0D 0A   : gzip,deflate..
130 : 41 63 63 65 70 74 2D 43 68 61 72 73 65 74 3A 20   Accept-Charset:
140 : 49 53 4F 2D 38 38 35 39 2D 31 2C 75 74 66 2D 38   ISO-8859-1,utf-8
150 : 3B 71 3D 30 2E 37 2C 2A 3B 71 3D 30 2E 37 0D 0A   ;q=0.7,*;q=0.7..
160 : 4B 65 65 70 2D 41 6C 69 76 65 3A 20 31 31 35 0D   Keep-Alive: 115.
170 : 0A 43 6F 6E 6E 65 63 74 69 6F 6E 3A 20 6B 65 65   .Connection: kee
180 : 70 2D 61 6C 69 76 65 0D 0A 43 6F 6E 74 65 6E 74   p-alive..Content
190 : 2D 4C 65 6E 67 74 68 3A 20 31 31 35 0D 0A 43 6F   -Length: 115..Co
1a0 : 6E 74 65 6E 74 2D 54 79 70 65 3A 20 61 70 70 6C   ntent-Type: appl
1b0 : 69 63 61 74 69 6F 6E 2F 6F 63 73 70 2D 72 65 71   ication/ocsp-req
1c0 : 75 65 73 74 0D 0A 0D 0A 30 71 30 6F 30 4D 30 4B   uest....0q0o0M0K
1d0 : 30 49 30 09 06 05 2B 0E 03 02 1A 05 00 04 14 6C   0I0...+........l
1e0 : 2B C5 5A AF 8D 96 BF 60 AD F8 1D 02 3F 23 B4 8A   +.Z....`....?#..
1f0 : 00 59 C2 04 14 A5 EF 0B 11 CE C0 41 03 A3 4A 65   .Y.........A..Je
200 : 90 48 B2 1C E0 57 2D 7D 47 02 10 59 E1 92 59 1F   .H...W-}G..Y..Y.
210 : 93 4D 7A DE CC 94 6F 92 4C 79 E2 A2 1E 30 1C 30   .Mz...o.Ly...0.0
220 : 1A 06 09 2B 06 01 05 05 07 30 01 04 04 0D 30 0B   ...+.....0....0.
```

Figure 29: Bare Byte Unicode Decoding Attack screenshots.

### 3- HTTP Inspect Oversize Request URL Directory

This attempt will trigger whenever the HTTP-Inspect pre-processor detects a request for a URL that is longer than a specified length, which violates the HTTP handler policy. This may indicate an attack or an attempt to evade an IDS. Web servers are reported prone to a Denial of Service condition when a long request is made to the server using Unicode characters. The HTTP-Inspect pre-processor will generate this event since a Domino server vulnerable and can be attacked in this way. Specifically, when a request is made to /cgi-bin/ with approximately 330 Unicode characters appended to the URL, the web server will crash and a DoS condition will be evident. Stack-based buffer overflow in the map URL function for Apache Tomcat JK Web Server Connector 1.2.19 and 1.2.20, as used in Tomcat 4.1.34 and 5.5.20, allows remote attackers to execute arbitrary code via a long URL that triggers the overflow in a URI. The maximum expected length of the URL could be user configured. To mitigate this terrified incident by controlling the HTTP Inspect configuration options properly. An attacker may supply an over-long URI in an attempt to evade IDS and perform a successful attack against a web server. [36]

| HTTP Inspect Oversize Request | Analysis |
|---|---|
| Kind of Alert | True positive |
| Root cause | Generated automatically. Based on my second screenshot, I highlighted the windows size; which is approximately **32928**, with enough Unicode characters appended to the URL as the screen shot shown. |
| Percentage | 3% 6 alerts went off. |
| Whois Command | Various IP addresses. |



| | | | | | | |
|---|---|---|---|---|---|---|
| □ | #0-(1-41) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:55:52 | 192.168.5.37 | 66.220.149.18 | TCP |
| □ | #1-(1-95) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:24 | 192.168.5.247 | 74.125.65.149 | TCP |
| □ | #2-(1-96) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:25 | 192.168.5.247 | 157.166.226.208 | TCP |
| □ | #3-(1-97) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:25 | 192.168.5.247 | 157.166.226.208 | TCP |
| □ | #4-(1-98) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:25 | 192.168.5.247 | 157.166.226.208 | TCP |
| □ | #5-(1-99) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:25 | 192.168.5.247 | 157.166.226.208 | TCP |
| □ | #6-(1-100) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:25 | 192.168.5.247 | 74.125.65.149 | TCP |
| □ | #7-(1-101) | [snort] (http_inspect) OVERSIZE REQUEST-URI DIRECTORY | 2011-04-10 23:58:26 | 192.168.5.247 | 157.166.226.208 | TCP |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 192.168.5.37 | 66.220.149.18 | 4 | 20 | 0 | 1396 | 17022 | no | 0 | 64 | 21834<br>=<br>0x554a |

| | |
|---|---|
| Options | *none* |

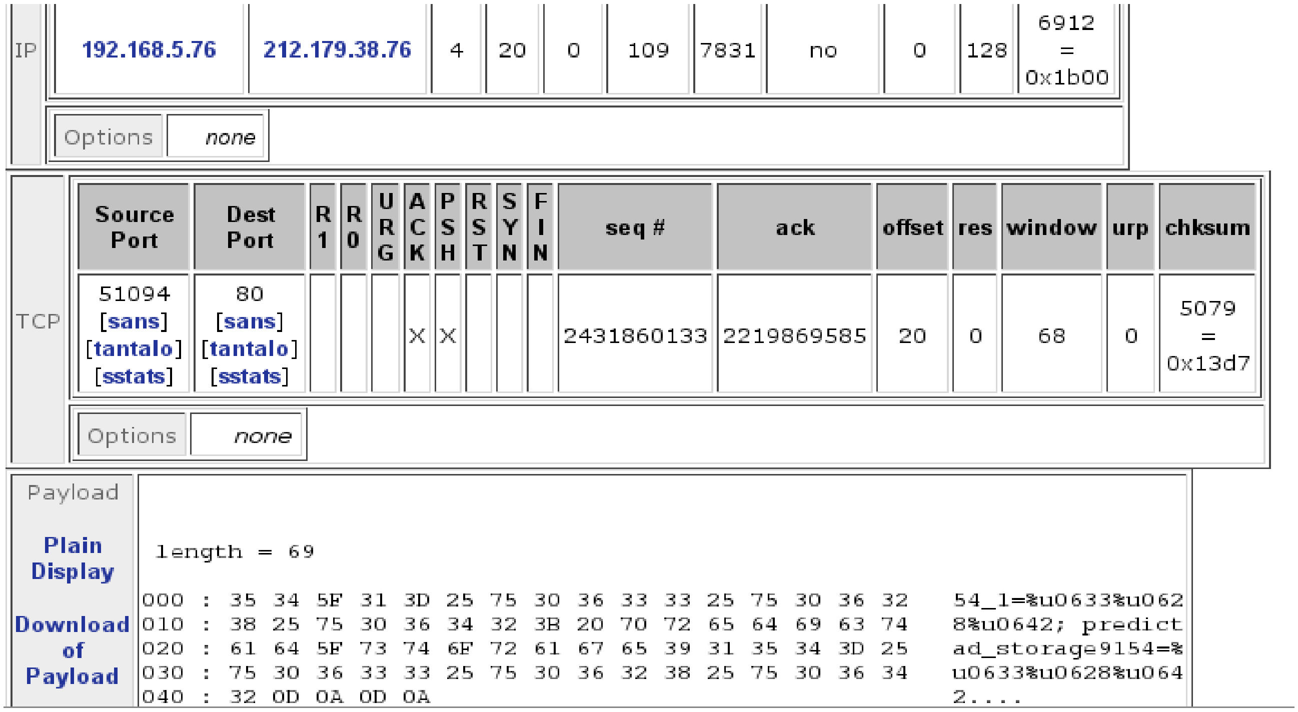
| Source Port | Dest Port | R 1 | R 0 | U R G | A C K | P S H | R S T | S Y N | F I N | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 55767<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | | | | | 2014232162 | 2036781765 | 32 | 0 | 32928 | 0 | 22697<br>=<br>0x58a9 |

Figure 30: HTTP Inspect Oversize Request URL Directory Screenshots.

### 4- HTTP-Inspect U Encoding

U Encoding attempt is generated when the pre-processor HTTP-Inspect detects network traffic that may constitute an attack. This event is generated when Unicode characters are present in a request sent to a web server. This may indicate an attempt to evade an IDS in an attempted attack against the server. No known browsers use Unicode encoding; it is likely that this event indicates a malicious request. Some attackers have the ability to encode malicious requests to the web server using Unicode characters, this may then evade an IDS monitoring traffic and an attacker could then launch a successful attack without being detected. As a corrective action, check the target host for signs of compromise. [37]

| HTTP U Encoding | Analysis |
|---|---|
| Kind of Alert | Unknown, we should leave it on with low priority tag. |
| Root cause | Generated manually |
| Percentage | 0% 2 alerts went off |
| Whois Command | Internal to BEZEQINT HOSTMASTERS TEAM in Israel. |



| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 192.168.5.76 | 212.179.38.76 | 4 | 20 | 0 | 109 | 7831 | no | 0 | 128 | 6912<br>=<br>0x1b00 |

| Options | none |
|---|---|

TCP

| Source Port | Dest Port | R 1 | R 0 | U R G | A C K | P S H | R S T | S Y N | F I N | seq # | ack | offset | res | window | urp | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 51094<br>[sans]<br>[tantalo]<br>[sstats] | 80<br>[sans]<br>[tantalo]<br>[sstats] | | | | X | X | | | | 2431860133 | 2219869585 | 20 | 0 | 68 | 0 | 5079<br>=<br>0x13d7 |

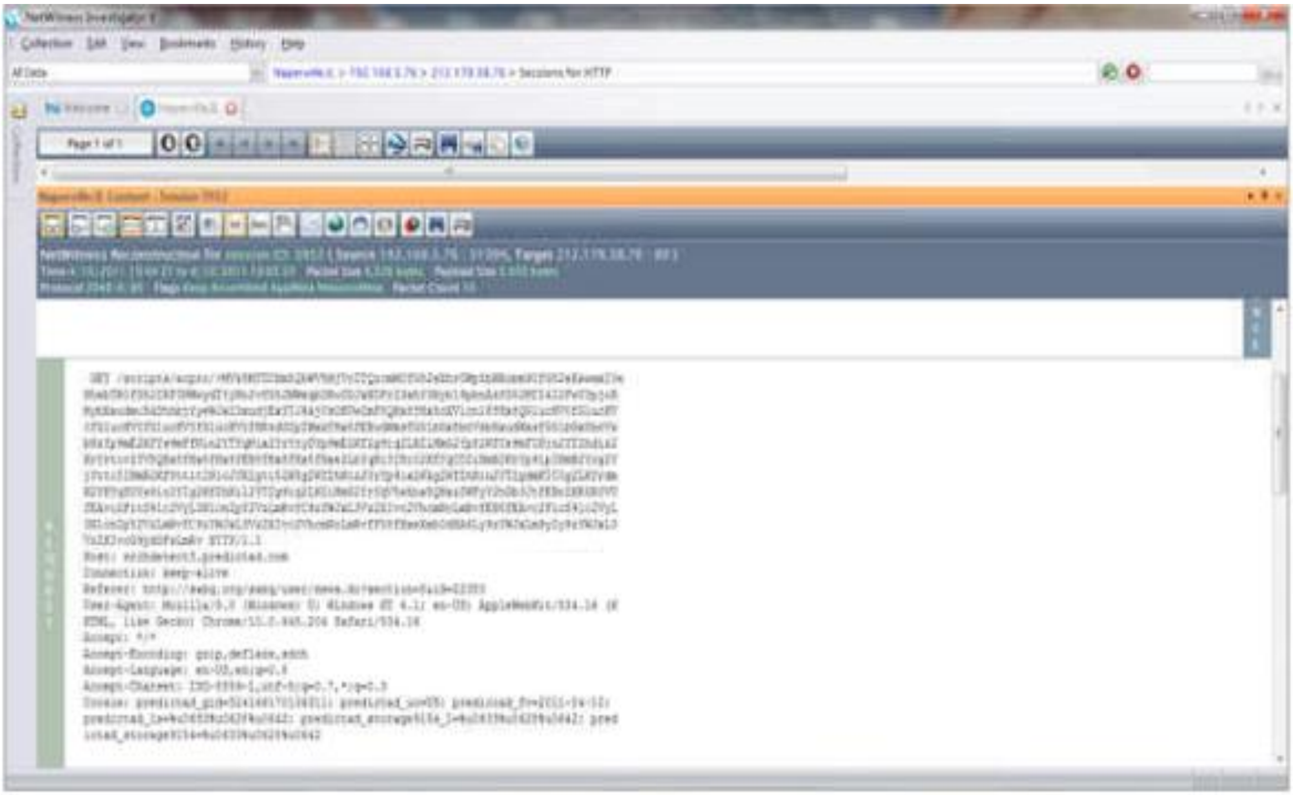
| Options | none |
|---|---|

Payload

Plain Display

Download of Payload

```
 length = 69

000 : 35 34 5F 31 3D 25 75 30 36 33 33 25 75 30 36 32   54_1=%u0633%u062
010 : 38 25 75 30 36 34 32 3B 20 70 72 65 64 69 63 74   8%u0642; predict
020 : 61 64 5F 73 74 6F 72 61 67 65 39 31 35 34 3D 25   ad_storage9154=%
030 : 75 30 36 33 33 25 75 30 36 32 38 25 75 30 36 34   u0633%u0628%u064
040 : 32 0D 0A 0D 0A                                     2....
```



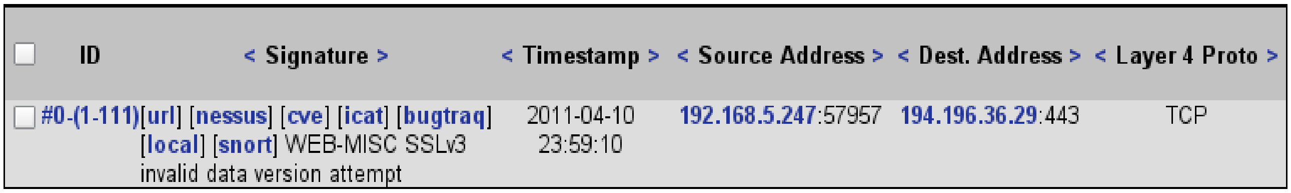
```
GET /scripts/[illegible] HTTP/1.1
Host: [illegible].predictad.com
Connection: keep-alive
Referer: http://[illegible]
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US) AppleWebKit/534.16 (KHTML, like Gecko) Chrome/10.0.648.204 Safari/534.16
Accept: */*
Accept-Encoding: gzip,deflate,sdch
Accept-Language: en-US,en;q=0.8
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3
Cookie: predictad_[illegible]; predictad_[illegible]; predictad_[illegible];
predictad_[illegible]; predictad_storage[illegible]; pred
ictad_storage[illegible]
```

Figure 31: HTTP-Inspect U Encoding Attempt Screenshots.

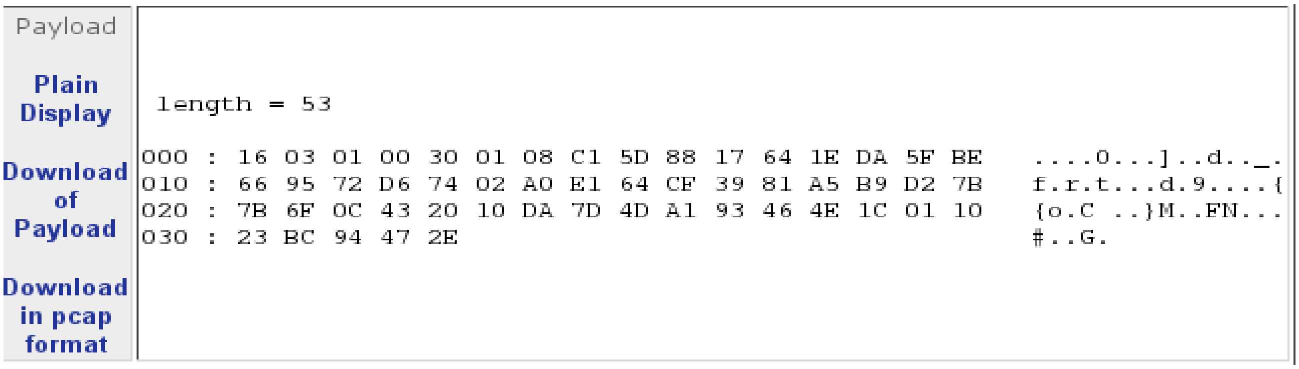
### 5- WEB-MISC SSLv3 invalid data version attempt

Web SSLv3 invalid data version attempt is made to exploit a known vulnerability in the Microsoft implementation of SSL Version 2.Classtype is attempting DOS. The vulnerability exists in the handling of SSL Version 2 requests that can be manipulated to cause a DoS condition in various software implementations used on Microsoft operating systems. The condition exists because of poor error handling routines in the Microsoft Secure Sockets Layer (SSL) library. SSL requests containing an invalid field, sent to vulnerable systems can cause the affected host to stop handling any further requests. Most commonly affected systems are Microsoft Windows 2000, 2003 and XP systems using SSL. An attacker needs to make an SSL request to an affected system that contains an invalid field. [38,39]

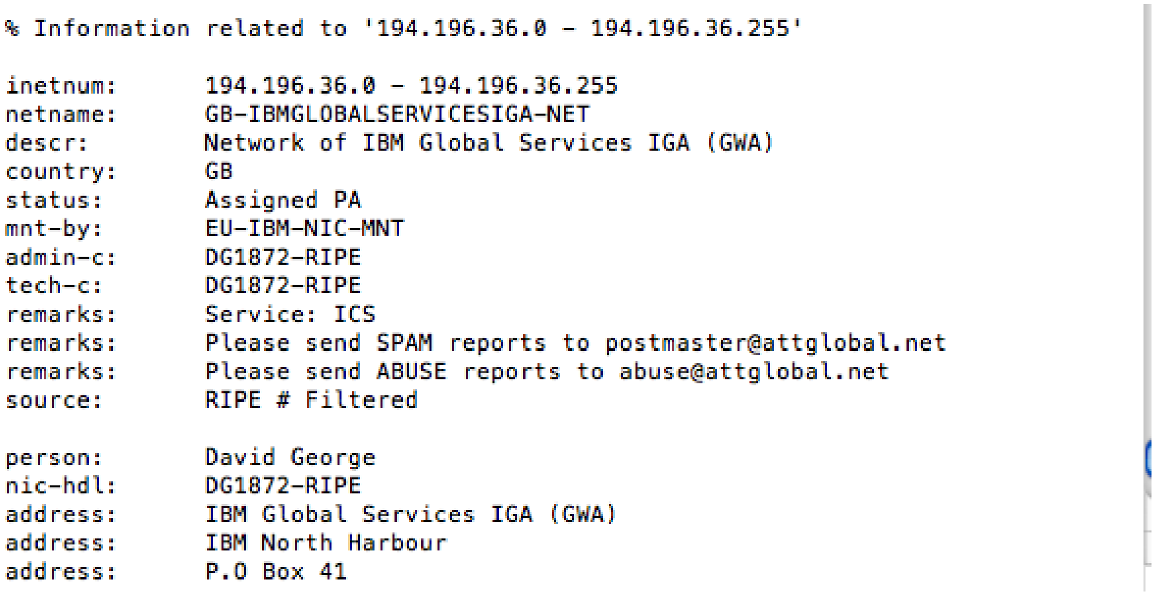
| SSLv3 invalid data version attempt | Analysis |
|---|---|
| Kind of Alert | Unknown. Check the targeted host for any sign. |
| Root cause | Generated automatically by a tool. |
| Percentage | 0% only one time alert went off. |
| Whois Command | From private IP to IBM Corp. |



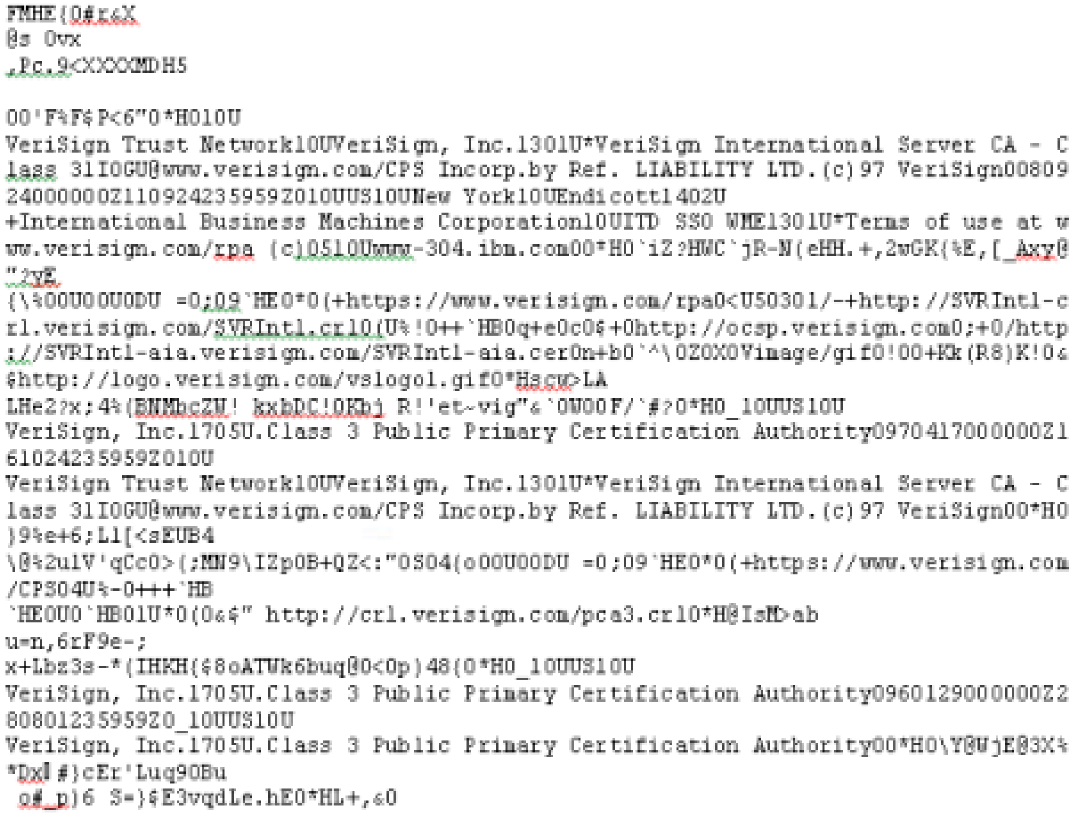
| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-111) | [url] [nessus] [cve] [icat] [bugtraq] [local] [snort] WEB-MISC SSLv3 invalid data version attempt | 2011-04-10 23:59:10 | 192.168.5.247:57957 | 194.196.36.29:443 | TCP |



```
length = 53

000 : 16 03 01 00 30 01 08 C1 5D 88 17 64 1E DA 5F BE   ....0...]..d.._.
010 : 66 95 72 D6 74 02 A0 E1 64 CF 39 81 A5 B9 D2 7B   f.r.t...d.9....{
020 : 7B 6F 0C 43 20 10 DA 7D 4D A1 93 46 4E 1C 01 10   {o.C ..}M..FN...
030 : 23 BC 94 47 2E                                     #..G.
```



```
% Information related to '194.196.36.0 - 194.196.36.255'

inetnum:        194.196.36.0 - 194.196.36.255
netname:        GB-IBMGLOBALSERVICESIGA-NET
descr:          Network of IBM Global Services IGA (GWA)
country:        GB
status:         Assigned PA
mnt-by:         EU-IBM-NIC-MNT
admin-c:        DG1872-RIPE
tech-c:         DG1872-RIPE
remarks:        Service: ICS
remarks:        Please send SPAM reports to postmaster@attglobal.net
remarks:        Please send ABUSE reports to abuse@attglobal.net
source:         RIPE # Filtered

person:         David George
nic-hdl:        DG1872-RIPE
address:        IBM Global Services IGA (GWA)
address:        IBM North Harbour
address:        P.O Box 41
```



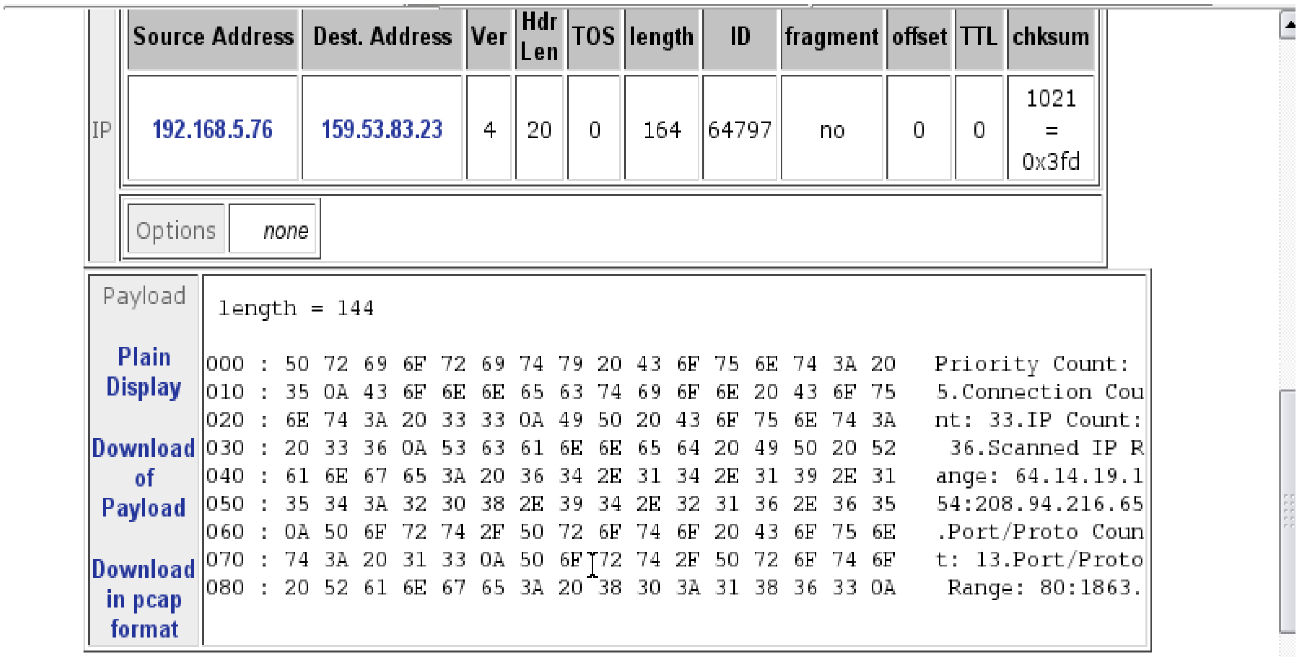
```
FMHE(0#r&X
@s 0vx
,Pc.9<XXXXMDH5

00'F%F$P<6"0*H010U
VeriSign Trust Network10UVeriSign, Inc.1301U*VeriSign International Server CA - C
lass 31I0GU@www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign00809
24000000Z110924235959Z010UUS10UNew York10UEndicott1402U
+International Business Machines Corporation10UITD SSO WHE1301U*Terms of use at w
ww.verisign.com/rpa (c)0510Uwww-304.ibm.com00*H0`iZ?HWC`jR-N(eHH.+,2wGK(%E,[_Axy@
"?yE
{\%00U00U0DU =0;09`HE0*0(+https://www.verisign.com/rpa0<U50301/-+http://SVRIntl-c
rl.verisign.com/SVRIntl.crl0(U%!0++`HB0q+e0c0$+0http://ocsp.verisign.com0;+0/http
://SVRIntl-aia.verisign.com/SVRIntl-aia.cer0n+b0`^\0Z0X0Vimage/gif0!00+Kk(R8)K!0&
$http://logo.verisign.com/vslogo1.gif0*Hscw>LA
LHe2?x;4%(BNMbcZW! kxbDC!0Kbj R!'et~vig"&`0W00F/`#?0*H0_10UUS10U
VeriSign, Inc.1705U.Class 3 Public Primary Certification Authority0970417000000Z1
61024235959Z010U
VeriSign Trust Network10UVeriSign, Inc.1301U*VeriSign International Server CA - C
lass 31I0GU@www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign00*H0
}9%e+6;L1[<sEUB4
\@%2ulV'qCc0>(;MN9\IZp0B+QZ<:"0S04(o00U00DU =0;09`HE0*0(+https://www.verisign.com
/CPS04U%-0+++`HB
`HE0U0`HB01U*0(0&$" http://crl.verisign.com/pca3.crl0*H@IsM>ab
u=n,6rF9e-;
x+Lbz3s-*(IHKH($8oATWk6buq@0<0p)48(0*H0_10UUS10U
VeriSign, Inc.1705U.Class 3 Public Primary Certification Authority0960129000000Z2
80801235959Z0_10UUS10U
VeriSign, Inc.1705U.Class 3 Public Primary Certification Authority00*H0\Y@WjE@3X%
*Dx▌#}cEr'Luq90Bu
 o#_p)6 S=}$E3vqdLe.hE0*HL+,&0
```

Figure 32: WEB-MISC SSLv3 invalid data version attempt Screenshots.

### 6- TCP Port Sweep

TCP port scan is generated when the sfPortscan pre-processor detects network traffic that may constitute an attack. This is normally an indicator of possible network reconnaissance and may be the prelude to a targeted attack against the targeted systems. A port scan is often the first stage in a targeted attack against a system. An attacker can use different port scanning techniques and tools to determine the target host operating system and application versions running on the host to determine the possible attack vectors against that host. An attacker often uses a port scanning technique to determine operating system type and version and also application versions to determine possible effective attack vectors that can be used against the target host. This is can be generated by one of today most powerful port scanning tools such as Nmap, Nessus, and Netcat. [40]

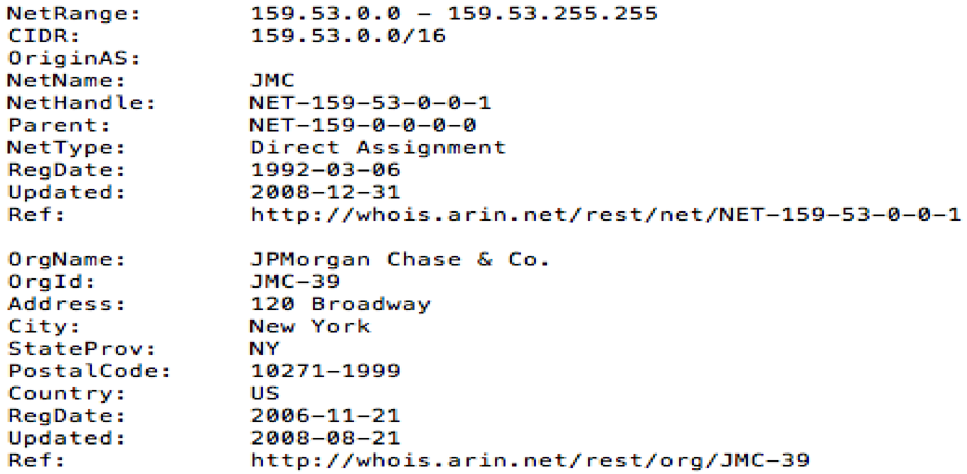
| TCP Port Sweep | Analysis |
|---|---|
| Kind of Alert | Very Serious threat, only Starbucks network administrator for security auditing or penetration test purposes can generate this kind of scan. |
| Root cause | Definitely manual (human act), I'm wondering who generated this scan to JP Morgan Chase Co. and why? |
| Percentage | 0% one alert went off |
| Whois Command | Internal scanner or attacker is somewhere next to me on Starbucks, scanned JPMorgan Chase, which is one of the oldest financial institutions in the United States. |



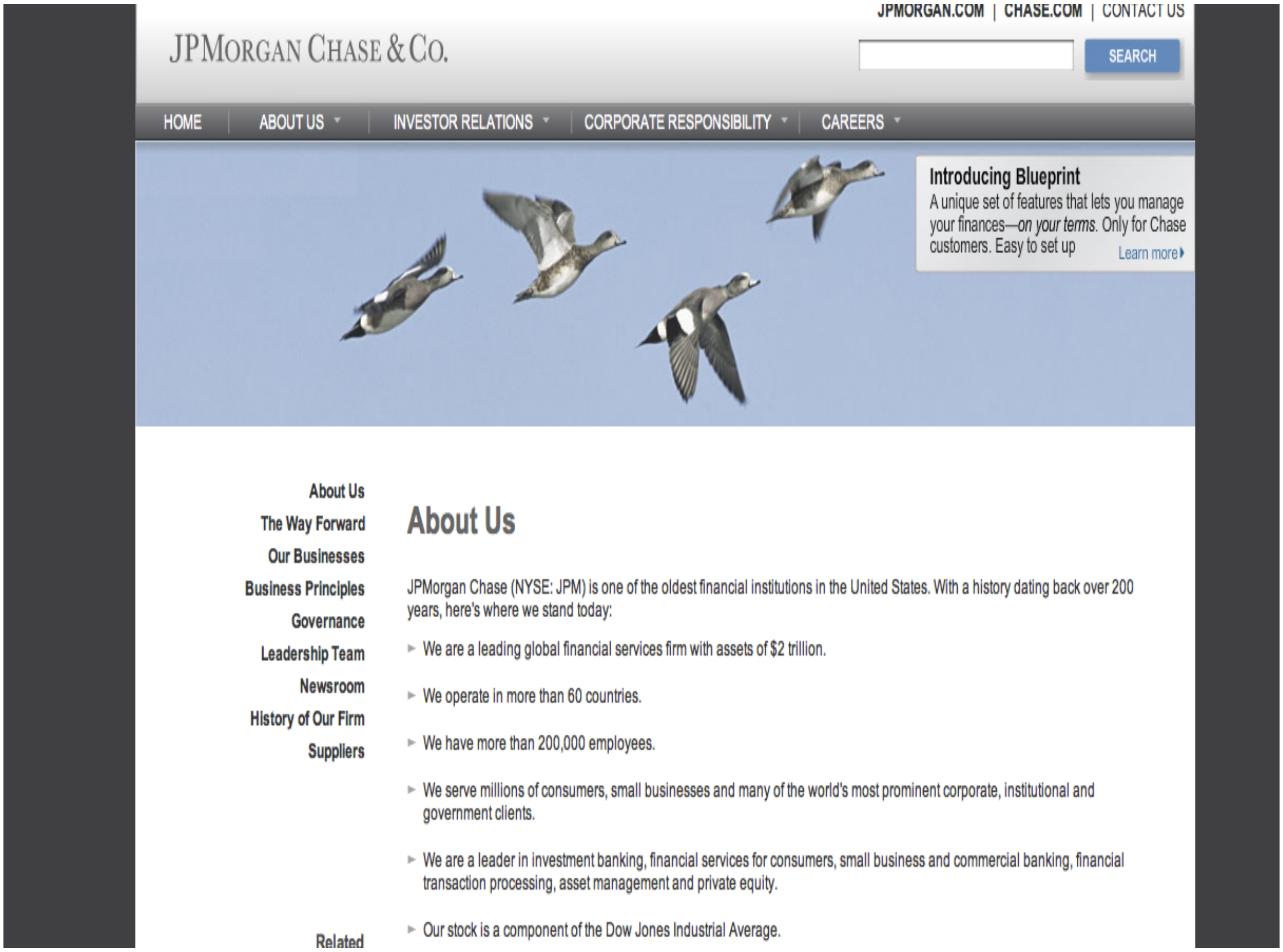
| | Source Address | Dest. Address | Ver | Hdr Len | TOS | length | ID | fragment | offset | TTL | chksum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 192.168.5.76 | 159.53.83.23 | 4 | 20 | 0 | 164 | 64797 | no | 0 | 0 | 1021 = 0x3fd |

| Options | none |
|---|---|

```
length = 144

000 : 50 72 69 6F 72 69 74 79 20 43 6F 75 6E 74 3A 20   Priority Count:
010 : 35 0A 43 6F 6E 6E 65 63 74 69 6F 6E 20 43 6F 75   5.Connection Cou
020 : 6E 74 3A 20 33 33 0A 49 50 20 43 6F 75 6E 74 3A   nt: 33.IP Count:
030 : 20 33 36 0A 53 63 61 6E 6E 65 64 20 49 50 20 52    36.Scanned IP R
040 : 61 6E 67 65 3A 20 36 34 2E 31 34 2E 31 39 2E 31   ange: 64.14.19.1
050 : 35 34 3A 32 30 38 2E 39 34 2E 32 31 36 2E 36 35   54:208.94.216.65
060 : 0A 50 6F 72 74 2F 50 72 6F 74 6F 20 43 6F 75 6E   .Port/Proto Coun
070 : 74 3A 20 31 33 0A 50 6F 72 74 2F 50 72 6F 74 6F   t: 13.Port/Proto
080 : 20 52 61 6E 67 65 3A 20 38 30 3A 31 38 36 33 0A    Range: 80:1863.
```



```
NetRange:       159.53.0.0 - 159.53.255.255
CIDR:           159.53.0.0/16
OriginAS:
NetName:        JMC
NetHandle:      NET-159-53-0-0-1
Parent:         NET-159-0-0-0-0
NetType:        Direct Assignment
RegDate:        1992-03-06
Updated:        2008-12-31
Ref:            http://whois.arin.net/rest/net/NET-159-53-0-0-1

OrgName:        JPMorgan Chase & Co.
OrgId:          JMC-39
Address:        120 Broadway
City:           New York
StateProv:      NY
PostalCode:     10271-1999
Country:        US
RegDate:        2006-11-21
Updated:        2008-08-21
Ref:            http://whois.arin.net/rest/org/JMC-39
```



**Introducing Blueprint**

A unique set of features that lets you manage your finances—*on your terms*. Only for Chase customers. Easy to set up

Learn more ▸

## About Us

JPMorgan Chase (NYSE: JPM) is one of the oldest financial institutions in the United States. With a history dating back over 200 years, here's where we stand today:

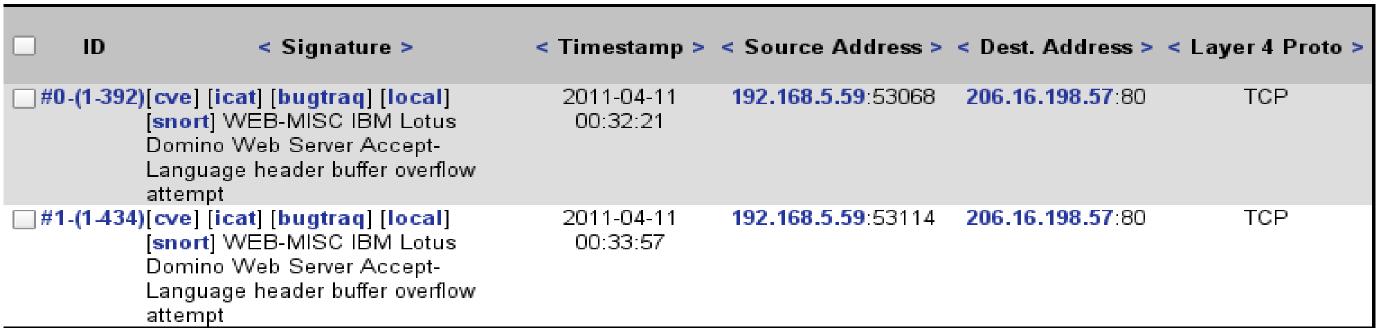
- We are a leading global financial services firm with assets of $2 trillion.
- We operate in more than 60 countries.
- We have more than 200,000 employees.
- We serve millions of consumers, small businesses and many of the world's most prominent corporate, institutional and government clients.
- We are a leader in investment banking, financial services for consumers, small business and commercial banking, financial transaction processing, asset management and private equity.
- Our stock is a component of the Dow Jones Industrial Average.

Figure 33: TCP Port Sweep attack screenshots.

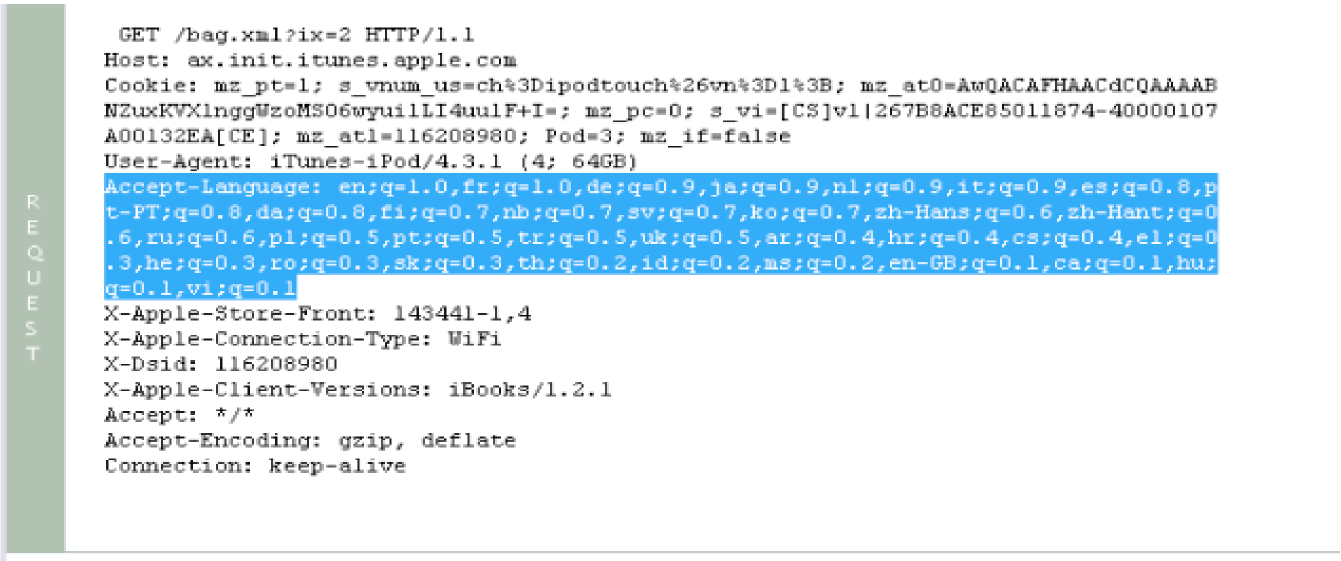
### 7- WEB-MISC IBM Lotus Domino Web Server Accept-Language header buffer overflow attempt

This event is generated when an attempt is made to exploit a known vulnerability in Lotus Domino. IBM-Long header can cause denial of service, information disclosure, loss of integrity, and complete administrator access. Stack-based buffer overflow in the Web Server service in IBM Lotus Domino before 7.0.3 FP1, and 8.x before 8.0.1, allows remote attackers to cause a denial of service (daemon crash) or possibly execute arbitrary code via a long Accept-Language HTTP header. [41, 42]

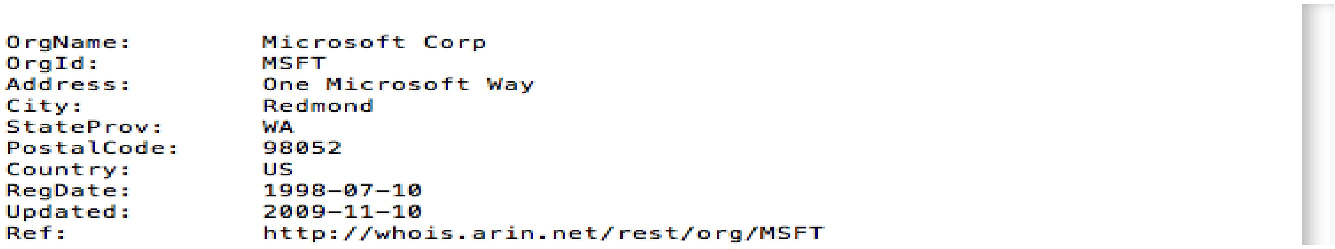
| Web Accept-Language header buffer overflow | Analysis |
|---|---|
| Kind of Alert | Serious, administrator should take a look at as an serious attack and make sure to block that IP address (192.168.5.59) |
| Root cause | Generated manually using script or malicious requests. |
| Percentage | 0% 2 alerts went off |
| Whois Command | Microsoft Web server which is using IBM Lotus server. |



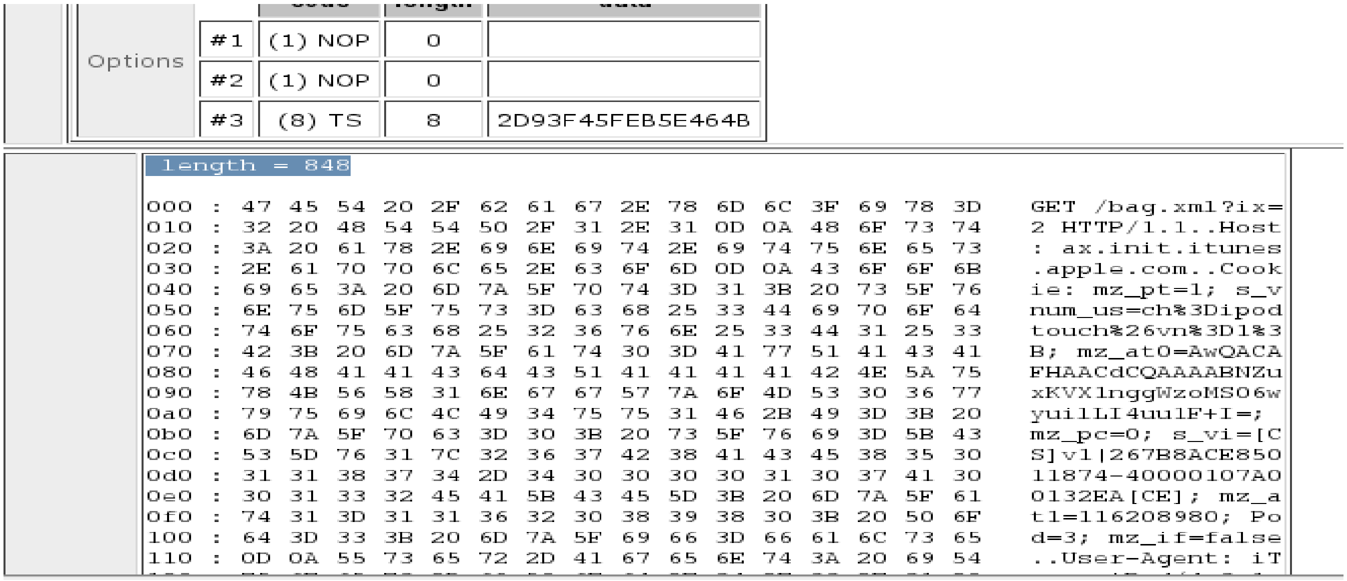
| □ | ID | < Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto > |
|---|---|---|---|---|---|---|
| □ | #0-(1-392) | [cve] [icat] [bugtraq] [local] [snort] WEB-MISC IBM Lotus Domino Web Server Accept-Language header buffer overflow attempt | 2011-04-11 00:32:21 | 192.168.5.59:53068 | 206.16.198.57:80 | TCP |
| □ | #1-(1-434) | [cve] [icat] [bugtraq] [local] [snort] WEB-MISC IBM Lotus Domino Web Server Accept-Language header buffer overflow attempt | 2011-04-11 00:33:57 | 192.168.5.59:53114 | 206.16.198.57:80 | TCP |



```
 GET /bag.xml?ix=2 HTTP/1.1
Host: ax.init.itunes.apple.com
Cookie: mz_pt=1; s_vnum_us=ch%3Dipodtouch%26vn%3D1%3B; mz_at0=AwQACAFHAACdCQAAAAB
NZuxKVXlnggWzoMSO6wyuilLI4uulF+I=; mz_pc=0; s_vi=[CS]v1|267B8ACE85011874-40000107
A00132EA[CE]; mz_atl=116208980; Pod=3; mz_if=false
User-Agent: iTunes-iPod/4.3.1 (4; 64GB)
Accept-Language: en;q=1.0,fr;q=1.0,de;q=0.9,ja;q=0.9,nl;q=0.9,it;q=0.9,es;q=0.8,p
t-PT;q=0.8,da;q=0.8,fi;q=0.7,nb;q=0.7,sv;q=0.7,ko;q=0.7,zh-Hans;q=0.6,zh-Hant;q=0
.6,ru;q=0.6,pl;q=0.5,pt;q=0.5,tr;q=0.5,uk;q=0.5,ar;q=0.4,hr;q=0.4,cs;q=0.4,el;q=0
.3,he;q=0.3,ro;q=0.3,sk;q=0.3,th;q=0.2,id;q=0.2,ms;q=0.2,en-GB;q=0.1,ca;q=0.1,hu;
q=0.1,vi;q=0.1
X-Apple-Store-Front: 143441-1,4
X-Apple-Connection-Type: WiFi
X-Dsid: 116208980
X-Apple-Client-Versions: iBooks/1.2.1
Accept: */*
Accept-Encoding: gzip, deflate
Connection: keep-alive
```



```
OrgName:        Microsoft Corp
OrgId:          MSFT
Address:        One Microsoft Way
City:           Redmond
StateProv:      WA
PostalCode:     98052
Country:        US
RegDate:        1998-07-10
Updated:        2009-11-10
Ref:            http://whois.arin.net/rest/org/MSFT
```



| | | | | |
|---|---|---|---|---|
| Options | #1 | (1) NOP | 0 | |
| | #2 | (1) NOP | 0 | |
| | #3 | (8) TS | 8 | 2D93F45FEB5E464B |

```
length = 848

000 : 47 45 54 20 2F 62 61 67 2E 78 6D 6C 3F 69 78 3D    GET /bag.xml?ix=
010 : 32 20 48 54 54 50 2F 31 2E 31 0D 0A 48 6F 73 74    2 HTTP/1.1..Host
020 : 3A 20 61 78 2E 69 6E 69 74 2E 69 74 75 6E 65 73    : ax.init.itunes
030 : 2E 61 70 70 6C 65 2E 63 6F 6D 0D 0A 43 6F 6F 6B    .apple.com..Cook
040 : 69 65 3A 20 6D 7A 5F 70 74 3D 31 3B 20 73 5F 76    ie: mz_pt=1; s_v
050 : 6E 75 6D 5F 75 73 3D 63 68 25 33 44 69 70 6F 64    num_us=ch%3Dipod
060 : 74 6F 75 63 68 25 32 36 76 6E 25 33 44 31 25 33    touch%26vn%3D1%3
070 : 42 3B 20 6D 7A 5F 61 74 30 3D 41 77 51 41 43 41    B; mz_at0=AwQACA
080 : 46 48 41 41 43 64 43 51 41 41 41 41 42 4E 5A 75    FHAACdCQAAAABNZu
090 : 78 4B 56 58 31 6E 67 67 57 7A 6F 4D 53 30 36 77    xKVX1nggWzoMS06w
0a0 : 79 75 69 6C 4C 49 34 75 75 31 46 2B 49 3D 3B 20    yuilLI4uu1F+I=;
0b0 : 6D 7A 5F 70 63 3D 30 3B 20 73 5F 76 69 3D 5B 43    mz_pc=0; s_vi=[C
0c0 : 53 5D 76 31 7C 32 36 37 42 38 41 43 45 38 35 30    S]v1|267B8ACE850
0d0 : 31 31 38 37 34 2D 34 30 30 30 30 31 30 37 41 30    11874-40000107A0
0e0 : 30 31 33 32 45 41 5B 43 45 5D 3B 20 6D 7A 5F 61    0132EA[CE]; mz_a
0f0 : 74 31 3D 31 31 36 32 30 38 39 38 30 3B 20 50 6F    t1=116208980; Po
100 : 64 3D 33 3B 20 6D 7A 5F 69 66 3D 66 61 6C 73 65    d=3; mz_if=false
110 : 0D 0A 55 73 65 72 2D 41 67 65 6E 74 3A 20 69 54    ..User-Agent: iT
```

Figure 34: IBM Lotus Web Server Accept-Language header buffer overflow attempt Screenshots.

## Recommendation of Server Cyber Threats

Figure 35 summarizes the recommended actions for each of the threats described in this chapter.

| Server Threats | Recommended Actions |
|---|---|
| Open Port Scan | Check for other events that targeting the host, compromise, and apply an appropriate patch. Also, need to block IP 192.169.5.76 & secure used ports and shut down unused ports. Also, I recommend leaving the alert ON with low priority, since it caused by Human scanned for open ports on two different other host organizations for open port both 80 & 443 |
| Bare Byte Unicode Decoding | Check the target host for signs of compromise. Apply any appropriate vendor supplied patches. Admin should take a look at theses http requests and check the server for any event. Also, I recommend setting a rule to silence drop packet if it Unintended human request, and violates RFC recommendations. If not, I'd leave it on with mid priority. |
| HTTP Inspect Oversize Request | Check the target host for signs of compromise. Apply any appropriate vendor supplied patches. Upgrade to the latest non-affected version of the software. |

| | |
|---|---|
| &<br>TCP Port sweep | |
| HTTP-Inspect U Encoding | Kind of Alert: False Positive "noise", it should be silently dropped, since it's known vulnerability and direct a log to admin<br>Root cause: based on time, I realized that attack is tool to make approximately 330 unicode characters appended to the URL as the screen shot sownApply any appropriate vendor supplied patches. This event can be controlled using the HTTP-Inspectconfiguration options. |
| WEB SSLv3 invalid data version attempt | Apply the appropriate vendor supplied patches. actually, this is my using side jacking tools to https to destination IBM, UK. I was able to trigger this alert |
| MISC IBM Lotus Domino WEB Server Accept-Language header buffer Overflow | Upgrade to the latest non-affected version of the software. Apply the appropriate vendor supplied patches. Serious, Admin should take a look at as an serious attack and make sure to block that IP address 192.168.5.59 |

# Chapter 7: Conclusion

To secure a network, it is essential to first define the threats that must be mitigated. Knowledge of these threats is important to understanding the reasons behind the various cyber-threats. As demonstrated, organizations should conduct risk assessments to identify the specific threats in advance against their security posture and determine the effectiveness of existing security controls in countering these threats. Consequently, the effective management of information technology resources is crucially important to any business that has public hotspot wifi. Because of the inherent nature of wireless communication, wireless networks require increased cooperation and coordination between network administrators and senior management.

The number of dimensions that make up each attack makes this measurement difficult. Nonetheless, it is possible to provide network administrators with a recommended action for each attack. This analysis is useful for any public hotspot wifi administrator. It was somewhat surprising to have found a few serious alerts on their network flowing without any detection software like an Intrusion Detection System. It is not hard to imagine how open wifi could be used by intruders and hackers to commit cyber-crimes and steal information right out of the air with little effort, no consequences, and walk away without detection. They use tools that are readily available on the Internet and can cause many problems for companies that do not take the time to understand the threats an unsecured wireless connection poses to their corporate network. By following the recommendations presented here, a wifi administrator can come to recognize the kinds of threats their system faces and how to counteract them.

# Resources:

1- Search security.In, (2009). Wireshark. Retrieved 2011, from http://searchsecurity.techtarget.in/definition/Wireshark

2- Thomas M. T. (Jul 16, 2004). *Wireless Security*. Retrieved from: http://www.informit.com/articles/article.aspx?p=177383&seqNum=5.

3- Roesch, M (July 14, 2011). *The Snort Project*. Retrieved Sep 26, 2011, from http://www.snort.org/assets/166/snort_manual.pdf

4- Roelker, D, Norton, M, & Hewlett, J (2004-09-08). *sfPortscan*. Retrieved 2011, from http://cvs.snort.org/viewcvs.cgi/snort/doc/README.sfportscan?rev=1.6

5- Netwitness, (2011). Interactive Threats Analysis, Investgator. Retrieved 2011, from http://www.netwitness.com/products-services/investigator

6- Intro to ICMP Gont, F (December 22, 2004). ICMP attacks against TCP. Retrieved 2011, from http://www.watersprings.org/pub/id/draft-gont-tcpm-icmp-attacks-03.txt

7- Snort, (2005). *ICMP Destination Unreachable Port Unreachable.* Retrieved from: http://www.snort.org/search/sid/402?r=1

8- SECURITY.OSDIR, ICMP PING *NIX documentation. Retrieved 09/22/2011, from http://osdir.com/ml/security.ids.snort.sigs/2003-07/msg00060.html

9- Snort, (2006). ICMP Ping BSD type. Retrieved from http://www.snort.org/search/sid/368
10- Sourcefire Vulnerability Research Team, (2006). Retrieved From: http://www.snort.org/search/sid/408

11- Sonicwall Internet Security, (2003). http://software.sonicwall.com/applications/ips/index.asp?ev=sig&sigid=352

12- ZyXL Security Policy, (2010-03-01). Retrieved From https://mysecurity.zyxel.com/mysecurity/jsp/policy.jsp?ID=1048942

13- Snort Library-SID 394. (2010). *Snort.* Retrieved from: http://www.snort.org/search/sid/394. Last accessed 14th Sep 2011.4-

14- Wild Packet, ICMP: Destination Unreachable. http://www.wildpackets.com/resources/compendium/tcp_ip/unreachable

15- Juniper Network, (2007). SHELLCODE: X86 SETUID 0 (TCP). Retrieved 2011, from https://services.netscreen.com/restricted/sigupdates/nsm-updates/HTML/SHELLCODE:X86:SETUID-0-TCP.html
16- Snort, (2010). SHELLCODE x86 inc ecx NOOP. Retrieved 2011, from

http://www.snort.org/search/sid/1394

17- Seclists.org, (13 Mar 2011). alert 1394 shellcode x86 inc ecx noop. Retrieved 04/08/2011, from http://seclists.org/snort/2011/q1/1098

18- Snort.ID, (2009). CHAT Yahoo Messenger Request. Retrieved 19/09/2011, from http://www.snortid.com/snortid.asp?QueryId=3692

19- Snort, (2006). WEB-PHP Pajax arbitrary command execution attempt. Retrieved 2011, from http://www.snort.org/search/sid/8734

20- Redteam, PAJAX Remote Code Injection and File Inclusion Vulnerability. Retrieved 2011, from http://www.redteam-pentesting.de/advisories/rt-sa-2006-001.txt

21- Technical Information, Instant Messenger Security. Retrieved 2011, from http://www.technicalinfo.net/papers/IMSecurity.htm

22- Snort, CHAT MSN Request. Retrieved 2011, from http://www.snort.org/search/sid/16525

23- Active State, (2011). HTTP Inspector. Retrieved 2011, from http://docs.activestate.com/komodo/4.4/http-insp.html

24- MSDN, HTTP Handlers and HTTP Modules Overview. Retrieved 2011, from http://msdn.microsoft.com/en-us/library/bb398986.aspx

25- SnortID.com, (2009). Double Decoding Attack . Retrieved 2001, from http://www.snortid.com/snortid.asp?QueryId=119%3A2

26- Roelker, D HTTP IDS Evasions Revisited. Retrieved 2011, from http://docs.idsresearch.org/http_ids_evasions.pdf

27- Snort, (2009). CGI Calendar Access. Retrieved 2011, from http://www.snortid.com/snortid.asp?QueryId=1%3A882

28- Snort, (2009). IMAP SSLv2 openssl get shared ciphers overflow attempt. Retrieved 2011, from http://www.snort.org/search/sid/8440

29- National Database Vulnerabilities, (09/28/2006). Buffer overflow in the SSL. Retrieved 2011, from http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2006-3738

30- Snort, (2009). IIS Unicode CODEPOINT Encoding. Retrieved 2011, from http://www.snortid.com/snortid.asp?QueryId=119%3A7

31- Selfsecurity, (2004). WEB-MISC handler access. Retrieved 2011, from http://www.selfsecurity.org/TrendMap/signature/jpn/35.htm

32- Snort, Oversize Chunk Encoding . Retrieved 08/2011, from http://www.snortid.com/snortid.asp?QueryId=119%3A16

33- Sourcefire, WEB-CGI icat access. Retrieved 2011, from http://www.snort.org/search/sid/1606?r=1

34- UCCS, Open Port Scan. Retrieved 2011, from http://cs.uccs.edu/~cs591/ids/snort/doc/signatures/122-20.txt

35- Snort, (July, 2011). *Unicode Decoding* . Retrieved Sep 26, 2011, from http://www.snort.org/search/sid/119-4 Inline Citation -- (Snort, July, 2011)

36- Snort, (2009). HTTP Inspect Oversize Request URL Directory . Retrieved 28/10/2011, from http://www.snort.org/search/sid/119-15

37- Snort, (2009). HTTP-Inspect U Encoding. Retrieved 09/09/2011, from http://www.snortid.com/snortid.asp?QueryId=119%3A3

38- Snort.id, (2099). WEB-MISC SSLv3 invalid data version attempt. Retrieved 10/28/2011, from http://www.snort.org/search/sid/1-3486

39- Tenable Network Security, Inc., (2004-2009). Microsoft Windows SSL Library. Retrieved 2011, from http://www.nessus.org/plugins/index.php?view=single&id=12204

40- Snort-Alert, TCP Portsweep. Retrieved 10/28/2011, from http://www.aldeid.com/wiki/Snort-alerts/portscan-TCP-Portsweep

41- Security Focus, (2010). IBM-HTTP Header Buffer Overflow Vulnerability. Retrieved 2011, from http://www.securityfocus.com/bid/29310/discuss

42- Snort, (2008). WEB-MISC IBM. Retrieved 2011, from http://www.snort.org/search/sid/13819

43- Scarfone, K, Jansen, W, & Tracy, M (July 2008). Guide to General Server Security. Retrieved 2011, from http://bruteforcestudyguide.com/essaywriting/apastyle.htm

44- SANS, (2011). Information Security Policy Templates. Retrieved 2011, from http://www.sans.org/security-resources/policies/

45- IATAC, (September 25, 2009). Intrusion Detection System. Retrieved 2011, from http://iac.dtic.mil/iatac/download/intrusion_detection.pdf

46- Lisa P. (December 12, 2007). *The Caffe Latte Attack.* Retrieved from: http://www.wi-fiplanet.com/tutorials/article.php/3716241.

47- Syngress.com, (11/8/2006). Wireless Sniffing with Wireshark. Retrieved 2011, from http://www.willhackforsushi.com/books/377_eth_2e_06.pdf

48- Gerald, C. (October 2009). *Introduction To Wireshark.* Retrieved from: http://wiresharkdownloads.riverbed.com/video/wireshark/introduction-to- wireshark/. Last accessed 4th, Aug 2011.