Analysis of Internet Backbone Traffic and Header Anomalies observed

Wolfgang John and Sven Tafvelin
Chalmers University of Technology
Email: {johnwolf,tafvelin}@chalmers.se

ABSTRACT

Abstract of what they done in the project

The dominating Internet protocols, IP and TCP, allow some flexibility in implementation, including a variety of optional features. To support research and further development of these protocols, it is crucial to know about current deployment of protocol specific features and accompanying anomalies. This work is intended to reflect the current characteristics of Internet backbone traffic and point out misbehaviors and potential problems. On 20 consecutive days in April 2006 bidirectional traffic was collected on an OC-192 backbone link. The analysis of the data provides a comprehensive summary about current protocol usage including comparisons to prior studies. Furthermore, header misbehaviors and anomalies were found within almost every aspect analyzed and are discussed in detail. These observations are important information for designers of network protocols, network application and network attack detection systems.[1]

Categories and Subject Descriptors

C.2.3 [**Network Operations**]: Network monitoring

General Terms

Measurement

Keywords

Internet Measurement, Traffic Analysis, Header Anomalies

1. INTRODUCTION

Today, the Internet has emerged as the key component for commercial and personal communication. One contributing factor to the still ongoing expansion of the Internet is its versatility and flexibility. Applications and protocols keep changing not only with time [1], but also within geographical locations. Unfortunately, this fast development has left

Why they have done this research.

little time or resources to integrate measurement and analysis possibilities into the Internet infrastructure. However, the Internet community needs to understand the nature of Internet traffic in order to support research and further development [2]. It is also important to know about current deployment of protocol specific features and possible misuse. This knowledge is especially relevant in order to improve the robustness of protocol implementations and network applications, since increasing bandwidth and growing numbers of Internet users also lead to increased misuse and anomalous behavior [3]. One way of acquiring better understanding is to measure and analyze real Internet traffic, preferably on highly aggregated links. The resulting comprehensive view is crucial for a better understanding of the applied technology and protocols and hence for the future development thereof. This is important for establishing simulation models [4] and will also bring up new insights for related research fields, such as network security or intrusion detection.

Recent work

A number of studies on protocol specific features have been published earlier, based on a variety of datasets. Thompson et al. [5] presented wide-area Internet traffic characteristics on data recorded on OC-3 traffic monitors in 1997, including figures about packet size distribution and transport protocol decomposition. McCreary et al. [1] provided a longitudinal analysis of Internet traffic based on data collected on an OC3 link of the Ames Internet exchange in 1999 to 2000. Fractions of fragmented traffic were presented and the usage of Path MTU Discovery was inferred based on the packet size distribution. Shannon et al. [6] studied frequency and cause of IP fragmented traffic on data collected on different WANs (100Mbit Ethernet, OC3 and OC12) in March 2001. Fraleigh et al. [7] analyzed traffic measurements from the Sprint IP backbone, based on a number of traces taken on different OC12 and OC48 links in 2001-2002. Pentikousis et al. [8] indirectly quantified deployment of TCP options based on traces with incomplete header information. The data was collected between October 2003 and January 2004 on a number of OC3 and OC12 links by the NLANR/PMA. In that paper, recent figures about packet size distributions were presented as well. Already earlier, Allman [9] presented observations about usage of TCP options within traffic from a particular webserver in a one and a half year period from 1998-2000. Finally, in his investigations about the evolution of transport protocols, Medina et al. [10] presented usage of TCP features like ECN (RFC 3168) based on passive measurements on a local webserver during two weeks in February 2004.

What is Path MTU discovery?

[1]This work was supported by SUNET, the Swedish University Network

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
IMC'07, August 24–26, 2007, San Diego, CA, USA.
Copyright 2007 ACM 978-1-59593-713-1/07/0008 ...$5.00.

whats different about his research and the previous ones?

Despite these existing studies, there is a need for further measurement studies [2, 11]. Continued analysis work needs to be done on updated real-world data in order to be able to follow trends and changes in network characteristics. Therefore, in this work we will consequently continue to analyze IP and TCP, as they are the most common protocols used in today's Internet, and compare the results to previous work. After describtion of the analyzed data in Section 2, we present our results for IP and TCP specific features in Section 3. Finally, Section 4 summarizes the main findings and draws conclusions. describing further sections

2. METHODOLOGY

What method did they use for this research and project?

2.1 Collection of Traces

The traffic traces have been collected on the outermost part of an SDH ring running Packet over SONET (PoS). The traffic passing the ring to (outgoing) and from (incoming) the Internet is primarily routed via our tapped links. This expected behavior is confirmed by SNMP statistics showing a difference of almost an order of magnitude between the tapped link and the protection link. Simplified, we regard the measurements to be taken on links between the region of Göteborg, including exchange traffic with the regional access point, and the rest of the Internet.

On the two OC-192 links (two directions) we use optical splitters attached to two Endace DAG6.2SE cards. The DAG cards captured the first 120 bytes of each frame to ensure that the entire network and transport header information is preserved. The data collection was performed between the 7th of April 2006, 2AM and the 26th of April 2006, 10AM. During this period, we simultaneously for both directions collected four traces of 20 minutes each day at identical times. The times (2AM, 10AM, 2PM, 8PM) were chosen to cover business, non-business and nighttime hours. Due to measurement errors in one direction at four occasions we have excluded these traces and the corresponding traces in the opposite direction.

2.2 Processing and Analysis

How to sanitize?

After storing the data on disk, the payload beyond transport layer was removed and the traces were sanitized and desensitized. This was mainly done by using available tools like Endace's *dagtools* and CAIDA's *CoralReef*, accompanied by own tools for additional consistency checks, which have been applied after each preprocessing step to ensure sanity of the traces. Trace sanitization refers to the process of checking and ensuring that the collected traces are free from logical inconsistencies and are suitable for further analysis. During our capturing sessions, the DAG cards discarded a total of 20 frames within 12 different traces due to receiver errors or HDLC CRC errors. Another 71 frames within 30 different traces had to be discarded after the sanitization process due to IP checksum errors.

By desensitization the removing of all sensitive information to ensure privacy and confidentiality is meant. The payload of the packets was removed earlier, so we finally anonymized IP addresses using the prefix preserving CryptoPAN [12]. After desensitization, the traces were moved to a central storage. An analysis program was run on the data to extract cummulated statistical data into a database. For packets of special interest, corresponding TCP flows have been extracted.

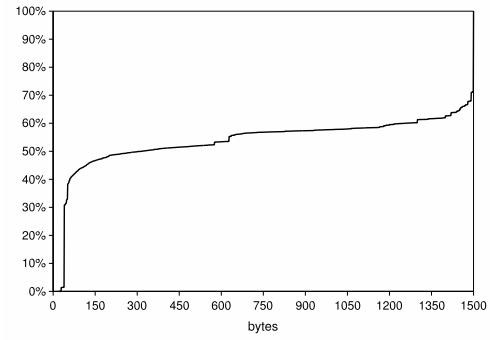

Figure 1: Cum. IPv4 Packet Size Distribution

3. RESULTS

The 148 traces analyzed sum up to 10.77 billion PoS frames, containing a total of 7.6 TB of data. 99.97% of the frames contain IPv4 packets, summing up to 99.99% of the carried data. The remaining traffic consists of different routing protocols (BGP, CLNP, CDP). The results in the remainder of this paper are based on IPv4 traffic only.

3.1 General Traffic Properties

Not trimodal but is bimodal as you can see in above

3.1.1 IP packet size distribution

In earlier measurements, cumulative distribution of IPv4 packet lengths was reported to be trimodal, showing major modes at small packet sizes just above 40 bytes (TCP acknowledgments), large packets around 1500 bytes (Ethernet MTU) and default datagram sizes of 576 bytes according to RFC 879. Data collected between 1997 and 2002 reported about fractions of default datagram sizes from 10% up to 40% [5, 1, 6, 7]. Pentikousis et al. [8] however showed in 2004, that packet size distribution was no longer trimodal, but rather bimodal, with default datagram sizes accounting for only 3.8% of all packets.

only has two peaks and not three peaks. In other words, the default sized packets are very low onlt 3.8%.

Fig. 1 illustrates the cumulative distribution of IPv4 packet lengths in our traces of 2006. The distribution is still bimodal, with the major portion of lengths between 40 and 100 bytes and between 1400 and 1500 bytes (44% and 37% of all IPv4 packets, resp.). The usage of the default datagram size of 576 byte was further decreased to a fraction of only 0.95%, now not even being among the first three most significant modes anymore. This is caused by the predominance of Path MTU Discovery in today's TCP implementations, which is confirmed later by the analysis of the IP flags and the TCP maximum segment size (MSS) option. On the other hand, two other notable modes appeared at 628 bytes and 1300 bytes, representing 1.76% and 1.1% of the IPv4 traffic, resp.

Is there a push set flag?

An analysis of TCP flows including a lot of 628 byte packets showed that these packets typically appear after full sized packets (MSS of 1460), often with the PUSH flag set. We suspect that they are sent by applications doing 'TCP layer fragmentation' on 2KB blocks of data, indicating the end of data a data block by PUSH. This is confirmed by flows where smaller MSS values have been negotiated (e.g. 1452). In this cases, the following packets became larger (e.g. 636 bytes) to add up to 2048 bytes of payload again. Examples for applications using such 2KB blocks for data transfer can

The destination of ports

be found in [13], where different file-sharing protocols using fixed block sizes are presented. A look at the TCP destination ports revealed that large fractions of this traffic are indeed sent to ports known to be used for popular file-sharing protocols like Bittorrent and DirectConnect. The notable step at 1300 bytes on the other hand could be explained by the recommended IP MTU for IPsec VPN tunnels [14].

Packets larger than 1500 bytes (Ethernet MTU) aggregate a fraction of 0.15%. Traffic of packets sized up to 8192 bytes was observed, but the major part (99.7%) accounts for packet sizes of 4470 bytes. A minor part of the >1500 byte sized packets represents BGP updates between backbone- or access routers. The majority of the large packet traffic (mainly 4470) could after thorough investigation be identified as customized data-transfer from a space observatory to a data center using jumbo-packets over Ethernet.

	2AM		10AM		2PM		8PM	
	Pkts	Data	Pkts	Data	Pkts	Data	Pkts	Data
TCP	91.3	97.6	91.5	96.8	93.2	97.1	91.4	97.2
UDP	8.5	2.3	7.6	2.8	6.1	2.7	8.3	2.7
ICMP	0.2	0.02	0.19	0.02	0.20	0.02	0.12	0.01
ESP	0.01	0.00	0.47	0.19	0.35	0.14	0.02	0.02
GRE	0.01	0.01	0.08	0.08	0.04	0.03	0.06	0.04

(a) IPv4 Protocol Breakdown (values in %)

OUTGOING UDP			
Date	Time	Packets	Data
2006-04-16	2PM	6.8	1.7
2006-04-16	8PM	40.6	5.1
2006-04-17	2AM	51.9	6.1
2006-04-17	10AM	58.1	7.1
2006-04-17	2PM	5.7	1.8

(b) UDP Burst (values in %)

Ip options, ECN, fragmentation how many packets?

Table 1: Transport Protocols

3.1.2 *Transport protocols*

The protocol breakdown in Table 1(a) once more confirms the dominance of TCP traffic. Compared to earlier measurements reporting about TCP accounting for around 90 - 95% of the data volume and for around 85-90% of IP packets, [5, 1, 6, 7], both fractions seem to be slightly larger in the analyzed SUNET data. In Table 1(a), the fractions of cumulated packets and bytes carried in the respective protocol are given in percent of total IPv4 traffic for the corresponding time.

Fraction meaning the number of packets!

An interesting observation can be made at the 2PM data. Here, the largest fraction of TCP and the lowest of UDP packets appear. A closer look at the differences between outgoing and incoming traffic revealed that three consecutive measurements on the outgoing link carried up to 58% UDP packets, not covering the 2PM traces, as shown in Table 1(b). These figures indicate a potential UDP burst of 14-24 hours of time. A detailed analysis showed that the packet length for the UDP packets causing the burst was just 29 bytes, leaving a single byte for UDP payload data. These packets were transmitted between a single sender and receiver address with varying port numbers. After reporting this network anomaly, the network support group of a University confirmed that the burst stemed from an UDP DoS script installed undetected on a webserver with a known vulnerability. Although TCP data was still predominant, a dominance of UDP packets over such a timespan could potentially lead to TCP starvation and raise serious concerns about Internet stability and fairness.

3.2 Analysis of IP Properties

3.2.1 *IP type of service*

Type of service

The TOS field can optionally include codepoints for Explicit Congestion Notification (ECN) and Differentiated Services. 83.1% of the observed IPv4 packets store a value of zero in the TOS field, not applying the mechanisms above. Valid 'Pool 1' DiffServ Codepoints (RFC 2474) account for 16.8% of all TOS fields.

TOS and differentiated services values

Medina et al. [10] reported about almost an doubling of ECN capable webservers from 1.1% in 2000 to 2.1% in 2004, but indicates that routers or middleboxes might erase ECT codepoints. In our data only 1.0 million IPv4 packets provide ECN capable transport (either one of the ECT bits set) and additionally 1.1 million packets actually show 'congestion experienced' (both bits set). This means that ECN is implemented in only around 0.02% of the IPv4 traffic. These numbers are consistent with the observations by Pentikousis et al. [8], suggesting that the number of ECN-aware routers is still very small.

3.2.2 *IP Options*

The analysis of IP options showed that they are virtually not used. Only 68 packets carrying IP options were observed. One 20-minute trace contained 45 packets with IP option 7 (Record Route) and 3 traces carried up to 12 packets with IP option 148 (Router Alert).

3.2.3 *IP fragmentation*

During the year 2000, McCreary et al. [1] observed an increase in the fraction IP packets carrying fragmented traffic from 0.03% to 0.15%. Indeed, one year later, Shannon et al. [6] reported fractions of fragmented traffic of up to 0.67%. Contrary to this trend, we found a much smaller fraction of 0.06% of fragmented traffic in the analyzed data. Even though these numbers are relatively small, there is still an order of magnitude difference between earlier and current results. 72% of the fragmented traffic in our data is transmitted during office hours, at 10AM and 2PM. We also observed that the amount of fragmented traffic on the incoming link is about 9 times higher than on the outgoing one.

While UDP and TCP are responsible for 97% and 3% respectively of all incoming fragmented segments, they just represent 19% and 18% of the outgoing. The remaining 63% of the outgoing fragmented traffic turned out to be IPsec ESP traffic (RFC 4303), observed between exactly one source and one receiver during working hours on weekdays. Each fragment series in this connection consists of one full length Ethernet MTU and one additional 72 byte fragment. This can easily be explained by an unsuitably configured host/VPN combination transmitting 1532 bytes (1572 - 40 bytes IP and TCP header) instead of the Ethernet MTU due to the additional ESP header. The dominance of UDP among fragmented traffic is not surprising, since Path MTU Discovery is a TCP feature only.

Fragmentation first packet

The first packets in all observed fragment series are in 96.7% sized larger or equal than 1300 bytes. This goes along with the assumption that fragments are sent in-order and the first segments should be full sized MTUs. It should be noted that 1.6% of first packets in fragment series are smaller than 576 bytes. This is not surprising, considering an earlier observation by Shannon et al. [6] that about 8% of fragment series are sent in reverse-order, sending the smallest segment first. This is accepted behavior, since the IP specification (RFC 791) does not prescribe any sizes of fragments. Another reason for small first segments are misconfigured networks or deliberate use of small MTUs, like serial links (RFC 1144) connected to the backbone. An example for such unusual small sized fragments of only 244 bytes will be given in the next subsection.

3.2.4 IP flags

The analysis of the IP flags (fragment bits) revealed that 91.3% of all observed IP packets have the don't fragment bit (DF) set, as proposed by Path MTU Discovery (RFC 1191). 8.65% use neither DF nor MF (more fragments) and 0.04% set solely the MF bit.

Following the IP specification (RFC 791) no other values are valid in the IP flag field. Nevertheless, we observed a total of 27,474 IPv4 packets from 70 distinct IP sources with DF and MF set simultaneously. About 35 of those invalid bit values are evenly observed among both directions in all traces, with exception of one burst of 21,768 packets in a trace of the incoming link. This burst stems from a 10 minutes long TCP flow between a local server on port 49999 and a remote client on the gaming port 1737 (UltimaD). Surprisingly, all the incoming traffic is fragmented to series of 244 byte long IP packets. The data carried by these fragment series adds up to full Ethernet MTUs size. Because being fragmented, each but the last fragment in a series has the MF bit set. Disregarding its actual fragmentation, each fragment also has the DF bit set. A similar behavior could be observed on the outgoing link, where one source generates 85% of all outgoing DF+MF packets, evenly distributed over 70 out of 76 measured times. Again, each IP packet has the DF bit set by default, while MF is set additionally when fragmentation is needed. Looking at the traffic pattern and considering that UDP port 53 is used, it seems to be obvious that there is a DNS server using improper protocol stacks inside the Göteborg region.

Reserved bit value

Additionally, we observed a total of 233 cases of a reserved bit with value 1, appearing in small numbers in most of the collected traces and stemming from 126 distinct sources. According to the IP standard (RFC 791) the reserved bit must be zero, so this behavior has to be regarded as misbehavior.

3.3 Analysis of TCP Properties

3.3.1 TCP Options

In an early study, Allman [9] reported about portions of hosts applying the Window Scale (WS) and Timestamp (TS) options, both increasing from about 15% to 20% during a 15 month period from 1998 to 2000. The SACK permitted option was shown to increase even further from 7% to 40%. No numbers for hosts applying the MMS option were given. The more recent approach to quantify TCP option deployment by Pentikousis et al. in 2004 [8] was unfortunately carried out on traces with incomplete header information. Since TCP option data was not available in these traces, their deployment had consequently to be analyzed indirectly. Our results, based on traces including complete header information, show that this indirect approach yielded quite accurate results.

Table 2(a) shows the deployment of the most important TCP options as fractions of the SYN and SYN/ACK segments, divided into summaries of the four times each day. The results show that MSS and SACK permitted options are widely used during connection establishment (on average 99.2% and 89.9% resp.). The positive trend of the SACK option deployment, as indicated by Allman, was obviously continued and the inferred values of Pentikousis et al. are finally confirmed. The frequent usage of the MSS option again indicates the dominance of Path MTU Discovery in TCP connections, since an advertised MSS is the precondition for this technique. The WS and TS options on the other hand are still applied to the same extent as in 2000 (17.9% and 14.5% resp.). In Table 2(b) the occurrence of

Kind	2AM	10AM	2PM	8PM
2(MSS)	99.0%	98.7%	99.7%	99.1%
3(WS)	21.4%	18.4%	16.6%	16.5%
4(SACK perm.)	91.0%	86.6%	88.9%	89.8%
8(TS)	18.2%	15.3%	13.3%	12.8%

(a) TCP Options in SYN segments

Kind	2AM	10AM	2PM	8PM
No Opt.	86.5%	85.2%	87.3%	88.6%
5(SACK)	3.1%	2.8%	2.9%	3.1%
8(TS)	9.7%	11.2%	9.0%	7.6%
19(MD5)	0.02%	0.02%	0.01%	0.01%

(b) TCP Options in all segments

Table 2: TCP Option Deployment

TCP options with respect to all TCP segments is summarized. Around 87% of the TCP segments do not carry any options at all. Only an average of 2.9% of all segments actually applies the SACK opportunity, which was permitted by around 90% of all connections. It is interesting, that although 15.5% of the connection establishments advertise usage of the TS option, it just reappears in 9.3% of all segments. This might be caused by TCP servers not responding with the TS option set in their initial SYN/ACK. All other option kinds were observed with very low frequency.

3.3.2 TCP option values

Allman [9] reported about 90% of connections advertising an MSS of about 1460 bytes in the SYN segment, leaving 6% for larger MSS values, and another 5% for MSS values of about 500 bytes. An analysis of advertised values within the MSS option field in our data revealed that the major portion (93.7%) of the MSS values still lies between 1400-1460 bytes, thus close to the Ethernet maximum (1500-40 byte for IP and TCP headers). Values larger than 1460 bytes are carried by only 0.06% of the MSS options, with values up to the maximum of 65535. Values smaller than 536 bytes (the default IP datagram size minus 40) are carried by another tiny fraction (0.05%), including MSS values down

to zero. The 53,280 packets carrying small MSS values are send by 2931 different IP addresses. The major fraction of the <536 MSS values carries a value of 512 (87.5%), followed by 64 (2.4%) and 260 (1.3%). Values down to 265 bytes can be explained by standard active fingerprinting attacks, like nmap [15], whereas smaller values are more likely to be DoS exploits.

In Allman's data from 2000, Window Scale (WS) factors as high as 12 appeared, with zero as the main factor, accounting for 84%, followed by a factor of one with about 15%. In our contemporary data, WS factor values appear in the range of 0 to 14. The most common scale factor with 58% is zero, which should not be interpreted as real factor, but as an offer to scale while scaling the own receive window by 1. The major real scale factor appears to be 2, with 30.8% deployment. Other scale factors in recognizable fractions are 3, 1, and 6, applied in respectively 4.2%, 4.1% and 1.0% of all segments carrying a WS option. As a general observation, the WS option is applied much more effective now, most probably due to bandwidth increases and larger data transfers. A detailed look at diurnal behavior of WS option values revealed that traces at nighttime (2AM) carry constantly about 10% more scale factor values of 2, compensated by around 10% less factors of zero.

how is the WS option implemented?

3.3.3 TCP option misbehavior

Connected to the analysis of TCP options, a couple of anomalies were encountered (Table 3(a)). The table shows only counts of packets, since the relative fractions are too small compared to the amount of total TCP segments. It should be mentioned that the differences between outgoing and incoming traffic lie typically in the order of a magnitude. Also diurnal differences can be observed, with non-working hours (2AM and 8PM) responsible for 67% of all reported anomalies.

Anomaly	2AM	10AM	2PM	8PM
Undef.Kind	1062	507	413	388
Invalid OL	1200	399	915	3020
Invalid HL	71	528	130	119

(a) TCP options and header lengths

Anomaly	2AM	10AM	2PM	8PM
RST+SYN+FIN	8	35	11	15
RST+SYN	25	70	43	27
SYN+FIN	4	22	8	9
Zero Flags	32	78	86	90
RST+FIN	10200	10988	14320	16334

(b) TCP flags

Table 3: Anomalies in TCP Headers

The first misbehavior experienced was the occurrence of undefined option types. Out of the 8bit range for TCP option kinds, only 26 are defined. From the remaining types almost all (228) have been observed. 522 distinct sources sent the 2370 undefined options observed, with 85% appearing on the incoming link. One single source sent 42% of these packets during the 20 minutes duration of one measurement at 2AM. Usage of a single destination port and 8200 different destination hosts within a one network prefix clearly indicate a scanning attack, even though only a minor fraction (6%) of the scanning traffic actually showed undefined options. The malformed packets carried instead of {MSS, NOP, NOP, SACK perm.} the option sequence of {MSS, random byte, random byte, 0, 0}. It seems likely that it is indeed the scanning software which is buggy and generates occasional malformed packets.

Another inconsistency encountered are option headers appearing to be valid while carrying option lengths that do not correspond to the total header length in the regular TCP header. 98.2% of the 5534 cases happened on the incoming link, with two sources responsible for 45% and 22% of such anomalous headers. The first source adds a SACK option with constant pattern to the TCP header, declaring an option header length of 180 bytes. This source was observed at 4 different days. The second source applies valid TCP options including an MSS value of 1460 during connection setup in SYN/ACK packets. However, also in the proceeding data packets an option of type 2 (MSS) appears, but this time followed by zeros, and thereby consequently advertising an option length of zero. According to the traffic pattern this source was a webserver. In total, 259 unique sources of this anomaly have been identified.

Finally, 848 TCP segments advertising header length values of less than 20 bytes were generated by 184 distinct sources, probably being DOS exploits. Again, the major fraction (91.3%) was observed in incoming traffic. 81.5% of the invalid values advertised a TCP header of zero length. The remaining 18.5% are evenly distributed between the remaining possible length values (in multiples of 4). The main source of zero byte TCP headers sends 351 such packets during a period of at at least 20 minutes. 351 unique destinations for 351 packets indicate a scanning campaign, this time to some well-known source port numbers (21, 23, 110, 80, 8080).

3.3.4 TCP Flags

Analyzing the TCP flag field, 10,972 ECN-setup SYN packets and just 800 ECN-setup SYN/ACK segments (RFC 3168) have been observed. The small numbers are consistent with earlier observations by Medina et al. [10], where only 0.2% of tested web clients advertise ECN capabilities. In section 3.2.1 we identified around 2.1 million ECN capable IP packets. This indicates that the few ECN enabled TCP connections represent large flows.

The urgent flag (URG) was set in only 663 segments. The acknowledgment flag (ACK) on the other hand was set in 98.6% of all segments, which is expected, since theoretically only the initial SYN packets should not carry an ACK flag. The push bit (PSH) was enabled in 22.4% of all segments.

In Table 3(b) we present packet counts for unexpected combinations of connection flags. The four first-listed anomalies have been seen in packets sent by 56 distinct sources. Such inconsistencies can easily be generated by port scanning tools like nmap [15]. We can rule out T/TCP as reason for SYN+FIN packets, since none of the 43 packets carried CC options (RFC 1644). The most frequent anomaly is connection termination with both, FIN and RST flags set. This was seen in 51,842 segments, send by 7576 unique source IP addresses. All connection flag anomalies are spread quite evenly over all measurements, with no particular sources to stand out.

how many percentage of the packets are TCP?
how many of the fragmented packets are UDP?
the number of 628 packet lengths decreased?

4. SUMMARY AND CONCLUSIONS

In order to be able to present contemporary characteristics of Internet traffic, 148 traces of 20 minutes duration have been collected on a pair of backbone links in April 2006. The analysis revealed that IP options are virtually not applied and IP fragmentation is done to a minor extent (0.06%), with UDP accounting for most IP fragments. The latter observation stems from an increased employment of TCP Path MTU Discovery, which was shown to be even more dominating than reported earlier. Regarding packet size distribution, two findings should be noted. First, IP packet lengths of 628 bytes have become even more common than the default datagram size, with P2P traffic identified as likely source. Second, except for router traffic, jumbo packets are used for a single custom application only and are not seen otherwise. Even though these observations are limited to our measurements from a single point in the Internet, this summary about current behavior of network protocols helps to understand the influence of additional protocol features on Internet traffic and can contribute to an improvement of future simulation models.

Additionally, a number of anomalies and inconsistencies have been observed, serving as pointers to keep in mind for protocol and application developers. As one cause for the otherwise rare occurrence of IP fragmentation additional headers introduced by VPN have been identified, advising application developers to use smaller MSS values. Furthermore, one single long- duration UDP burst was observed while gathering protocol statistics. This was found to be an UDP DoS attack, undetected by the network management tools in operation. This indicates the need for continuous refinement of network monitoring policies. The magnitude of the burst also raises stability and fairness concerns, calling for addition of some kind of congestion control to UDP. Finally, several types of misbehaviors within IP and TCP headers have been discussed. The anomalies observed could be explained by three different causes:

- Buggy or misbehaving applications or protocol stacks
- Active OS fingerprinting [13]
- Network attacks exploiting protocol vulnerabilities

Even though all header anomalies observed are rare compared to the total number packets, their existence shows that developers need to carefully design implementations. Almost any possible inconsistency in protocol headers will appear eventually, thus network protocols and applications have to be designed and implemented as robust as possible, leaving no vulnerabilities. Percentage of header anomalies?

Since access to traffic on highly aggregated links is still uncommon for researchers working on network security, our results form valuable input to related research on topics like large scale intrusion detection or traffic filtering. Besides quantifying the occurrence of different header anomalies 'in the wild', the results yield explanations for the origins of these commonly observed inconsistencies. Not every malformed packet header is necessarily part of attacking traffic, as proven by the example of the DNS server setting invalid fragmentation bits, but can also be introduced by improper protocol stacks. This information can be relevant when refining rule-sets for traffic filters, as applied in firewalls or network intrusion detection systems. Furthermore, knowledge about the nature of header anomalies can be interesting for researchers or developers creating stress tests for routers and other network components.

5. ACKNOWLEDGMENTS

The authors want to thank Pierre Kleberger for his kind technical support and Tomas Olovsson for his valuable and constructive comments throughout the MonNet project.

6. REFERENCES

[1] S. McCreary and K. Claffy, "Trends in wide area ip traffic patterns - a view from ames internet exchange," CAIDA, San Diego Supercomputer Center, Tech. Rep., 2000.

[2] N. Brownlee and K. Claffy, "Internet measurement," *IEEE Internet Computing*, vol. 08, no. 5, pp. 30–33, 2004.

[3] A. Householder, K. Houle, and C. Dougherty, "Computer attack trends challenge internet security," *Computer*, vol. 35, no. 4, pp. 5–7, 2002.

[4] S. Floyd and E. Kohler, "Internet research needs better models," ser. Comput. Commun. Rev. (USA), vol. 33. Princeton, NJ, USA: ACM, 2003, pp. 29–34.

[5] K. Thompson, G. J. Miller, and R. Wilder, "Wide-area internet traffic patterns and characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, 1997.

[6] C. Shannon, D. Moore, and K. Claffy, "Beyond folklore: observations on fragmented traffic," *IEEE/ACM Transactions on Networking*, vol. 10, no. 6, pp. 709–20, 2002.

[7] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot, "Packet-level traffic measurements from the sprint ip backbone," *Network, IEEE*, vol. 17, no. 6, pp. 6–16, 2003.

[8] K. Pentikousis and H. Badr, "Quantifying the deployment of tcp options - a comparative study," *IEEE Communications Letters*, vol. 8, no. 10, pp. 647–9, 2004.

[9] M. Allman, "A web server's view of the transport layer," *SIGCOMM Comput. Commun. Rev.*, vol. 30, no. 5, 2000.

[10] A. Medina, M. Allman, and S. Floyd, "Measuring the evolution of transport protocols in the internet," *Computer Communication Review*, vol. 35, no. 2, pp. 37–51, 2005.

[11] A. Hussain, G. Bartlett, Y. Pryadkin, J. Heidemann, C. Papadopoulos, and J. Bannister, "Experiences with a continuous network tracing infrastructure," in *MineNet'05: ACM SIGCOMM workshop on Mining network data*. New York, NY, USA: ACM Press, 2005.

[12] J. Xu, J. Fan, M. Ammar, and S. Moon, "On the design and performance of prefix-preserving ip traffic trace anonymization," in *IMW '01: ACM SIGCOMM Workshop on Internet Measurement*. New York, NY, USA: ACM Press, 2001.

[13] T. Karagiannis, A. Broido, N. Brownlee, k claffy, and M. Faloutsos, "File-sharing in the internet: A characterization of p2p traffic in the backbone," UC Riverside, Tech. Rep., 2003.

[14] CiscoSystems, "Ipsec vpn wan design overview," Cisco Doc., 2006. [Online]. Available: http://www.cisco.com/univercd/cc/td/doc/solution/ipsecov.pdf

[15] Fyodor, "Nmap security scanner," 1998. [Online]. Available: http://insecure.org/nmap/index.html