Observations in E-Mail Performance

Joachim Charzinski

Siemens Information and Communication Networks – Hofmannstr. 51, D-81359 Munich, Germany
phone: +49.89.722.46803 – fax: +49.89.722.26877 – email: j.charzinski@ieee.org

Abstract

E-mail is one of the most popular applications on the Internet. Like most traditional Internet services, the transport of mails relies on the Transmission Control Protocol (TCP), allowing the applications to adapt to almost any available network bandwidth and packet loss rates. As most mail client software for residential users tends to view the transmission and reception of mail messages as a more or less *interactive* process, the delays for sending and receiving e-mails become an issue to the users.

In this paper, we show that a significant share of the latency both for sending and receiving e-mails is due to serial processing of commands, which can hardly be reduced by increasing bandwidths. Three long-time real-life Internet traces have been evaluated using application-level analysis. A second result is that heavy-tailed distributions are not only found as expected in the mail sizes but also in the duration of command exchanges before e-mails are transmitted and in the number of e-mails transmitted in an SMTP, POP3 or IMAP connection.

Keywords: Application Level Performance; E-Mail; Traffic Measurement; SMTP; POP3; IMAP; Packet Trace

1 Introduction

Electronic mail (e-mail) is one of the most popular services and one of the largest sources of traffic on the Internet. The Transmission Control Protocol (TCP) is used both for reliable communication and to adapt to the bandwidth available on a link. However, contrary to frequent belief, this does not imply that the e-mail is completely uncritical in terms of performance requirements.

On the user level, e-mail is a method for the *asynchronous* delivery of messages and documents to other users on the net[1]. This view is adequate for the level of communication between one user and another, and it is one reason for the popularity of the service. However, the mode of communication gets more *synchronous* at the interface between the users and their e-mail clients, especially in residential set-ups. This is due to three reasons:

- Users who pay time-based Internet access charges may be careful not to stay online longer than necessary and therefore wait for the process of sending or retrieving an e-mail to complete. (Of course, this task can be delegated to the mail client software.)
- Some e-mail clients effectively block access to the mailbox or even to the computer's graphical user interface while retreiving or transmitting messages.
- Many people wait for the "mail sent successfully" message to make sure that an e-mail has actually been accepted by the mail server, to exclude addressing problems or simply to be sure that the message has left the local computer and that it is safe to close the Internet access connection.

The interaction sequence for a residential user sending an e-mail using a time-charged dial-up Internet access is sketched in Figure 1. After composing a message offline, the user instructs the mail client to send the message to its destination. After the dial-in, which can have happened before, the message is transferred to a mail server via the Simple Mail Transfer Protocol (SMTP) [1]. The mail client may inform the user about the current state of the transfer, e.g. by indications like "looking up mail server", "contacting mail server", "sending mail message", "mail sent successfully" before the user knows that the message has been received by the next hop mail server and it is safe to close the Internet access connection.

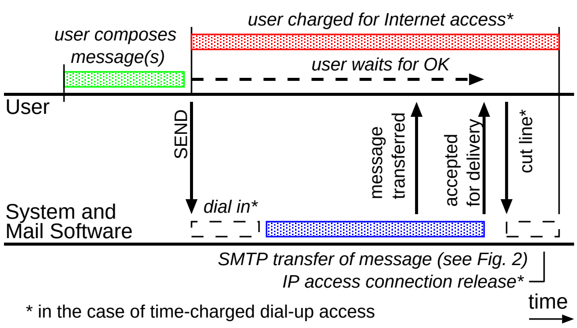

Figure 1: Interaction between user and mail software for sending an e-mail.

Due to the mixture of technical, financial and psychological reasons, the process of sending and retrieving e-mails is rather synchronous in nature for residential users and therefore the latencies observed with e-mail transfers are definitely a performance issue.

This paper is focused on the performance aspects of SMTP, POP3 and IMAP connections observed on the Internet. However, it must be mentioned that an increasing portion of users access their mailboxes via Web mail, using a Web browser interface and the HTTP protocol to write and read e-mails. As those connections cannot be distinguished from other Web traffic in anonymized

[1] But even the original Internet mail standard [1] views asynchronous and "instant" messaging as two realizations of the same concept.

packet header traces, it is impossible to extract performance results for Web mail. However, both the performance observed with general Web traffic [2, 3] and personal observations using Web mail indicate that the performance problems discussed here are not resolved but rather aggravated by the use of Web mail interfaces.

1.1 Contribution and Related Work

The main contribution of this paper is the observation of protocol performance for e-mail transfer protocols in real-life situations, based on an application level analysis of the transport layer data in packet header traces collected by passive monitoring. The method has been used before [2, 3] for HTTP performance analysis and is now employed to extract SMTP, POP3 and IMAP performance and traffic characteristics from long-term traffic traces.

Detailed results are gained for SMTP, POP3 and IMAP, describing protocol and traffic characteristics as well as the performance that was observed by the real users while the traffic traces were captured. Connection level traffic characteristics for several TCP based protocols have been investigated by Cáceres et al. [4] before without regard to application layer intra-connection details.

A similar analysis technique as used here has been applied by Smith et al. [5] who investigated Web traffic characteristics and their evolution over the years from unidirectional packet header traces.

Active measurement studies [6, 7, 8, 9], which have less relation to the performance perceived by real users but yield more reproducible results, have been conducted with a focus on HTTP. They have recognized latency as a main performance problem in Web access. Huitema and Weerahandi [10] used active measurements to investigate the delay due to DNS lookups.

1.2 Traces

Three extensive traffic traces are evaluated in this paper. Traces A and B have been captured at residential sites in Germany in 1998–1999. The set-up and basic characteristics are described in [11]. As the Internet and Internet applications have definitely evolved in the past years, a more recent trace available online from NLANR [12] was included as “Trace C” for comparison. A summary of the trace characteristics of the three traces is given in Table 1.

Trace A was collected during May through December 1998 from an Ethernet switch connecting around 100 students in their dorms to the Internet via ADSL, with the individual access lines configured to 2.5 Mbit/s downstream and 384 kbit/s upstream data rates. Due to technical constraints, users were divided into 15 groups, which were monitored cyclically for one week each.

Trace B was recorded for five weeks in March and April 1999 at a local Internet Service Provider giving Modem and ISDN dial-up access to around 300 users. The Internet backbone connection was a 128 kbit/s line; only a local HTTP and mail server could be reached at higher rates with data compression and ISDN channel bundling.

Table 1: Summary of the number of packets, connections and items observed in the traces for SMTP, POP3 and IMAP.

	Trace A	Trace B	Trace C
Packets	60 M	43 M	219 M
SMTP connections	2.1 k	3.4 k	335 k
SMTP mails	2.1 k	4.3 k	324 k
POP3 connections	34 k	31 k	88 k
POP3 mails	5 k	12.8 k	5.2 k
IMAP connections	–	–	6.3 k
IMAP mails	–	–	7.5 k

Trace C consists of the first seven days (Feb. 20–26, 2001) of the “Auckland-IV” trace [12] captured by the WAND research group [13] at the University of Auckland Internet uplink, available from NLANR MOAT [14].

1.3 Trace Evaluation

The measurement point on the access network between clients and core network allowed us to record the full traffic from and to users’ computers in Traces A and B. When considering delays, the additional delay between the measurement point and the client computer must be taken into account. This delay was found to be very small (at most a few ms) for the ADSL access in Trace A and rather large (more than 100 ms) on some modem lines of Trace B in a previous study [2]. Obviously, capturing traffic *behind* a local campus network as in Trace C reduces the observable associations to those between machines inside and outside the campus and the actual delay between the measurement point and a client computer is unknown. Therefore, measures were defined as round-trip measures where possible (e.g. in three-way handshakes), so that the actual allocation of delays between client, measurement point and server does not matter for those measures.

The packet based traces were evaluated by analyzing the time stamps, sizes and TCP flags of the recorded packets. Protocols and client/server roles were detected using the server side well-known port numbers for SMTP (25), POP3 (110) and IMAP (143).

In the analysis, a separate protocol state machine was run for each connection and protocol of interest, analyzing the state of the connection by observing the dialogue behavior between client and server through TCP flags and sizes of the packets exchanged. Some details of these state machines are given below for the different protocols.

1.3.1 DNS

The simple state machine for the Domain Name System (DNS) observed outgoing DNS requests from a client and waited for an incoming DNS packet directed to the same client port. Repeated request packets from the same port – signs of time-outs due to slow server reaction or packet loss – were ignored until a reply was received, i.e. the DNS lookup latency was measured relative to the instant of the original request in this case. A DNS request/reply pair was associated with the following TCP connection opened to a server if the client had not communicated with that server's IP address during the previous 15 minutes. Communication between DNS servers (port 53 to port 53, TCP or UDP) was ignored.

1.3.2 SMTP

The SMTP, POP3 and IMAP state machines relied solely on the timestamps, packets size, IP and TCP header information recorded in the traces. Evaluating packet size and TCP "Push" flag information allowed to extract characteristic events for these application layer protocols.

After the client opens an SMTP connection, the first downstream packet carrying data (packet size greater than 44 Bytes and the TCP "Push" flag set) is the server greeting. A dialogue of commands and answers takes place after the greeting. Before an SMTP client transmits an e-mail to the server, it sends the `DATA` command in a packet with the TCP "Push" flag set and 6 octets of payload data (`D`, `A`, `T`, `A` and two control octets). A unidirectional flow of data from the client to the server that follows this command can be safely recognized as an e-mail message – the server only acknowledges the data packets received from the client and does not send application layer data itself. The first packet from the server that contains application layer data after such a phase indicates the end of the e-mail transfer. Before starting a new mail upload to the server, a client would either close the connection and open a new one or send an SMTP `RSET` command (another packet with six octets of application layer data) to reset the application layer state of the connection.

The corresponding state machine implemented to analyze the traces contains some additional code to deal with the most common pattern variations which an expert could recognize by optical examination of a packet header trace. Around 1 % of the connections had to be dropped from the evaluation because they contained patterns which by human examination could not be recognized as making sense.

1.3.3 POP3 and IMAP

In general, the same state machine concept was used for POP3 and IMAP as for the SMTP analysis.

In contrast to SMTP connections, POP3 and IMAP connections do not necessarily transport e-mail messages as the client cannot know if there are new messages available before opening the connection. Correspondingly, the application layer communication patterns in POP3 and IMAP connections are more variable than in SMTP.

Mail messages – at least those transmitted in more than one TCP segment – were recognized by the server transmitting more than one data packet to the client and the client sending acknowledgment packets only without application layer data. This method for detecting e-mails is unable to distinguish between e-mails and long mailbox contents listings retrieved from the server as a result of a POP3 `LIST` or an IMAP `LIST` or `SEARCH` command. A proper distinction between e-mails and mailbox listings would require application layer information to be captured in the traces. Due to privacy concerns, this was not done in Traces A, B and C.

2 Sending E-Mails (SMTP)

In the typical residential scenario, the mail client program transmits an e-mail message to the Internet service provider's mail server using the Simple Mail Transfer Protocol (SMTP) described in RFC 821 [1]. As most residential users do not always keep their computers online or do not have fixed Internet addresses, mails are not delivered to these computers with SMTP but need to be stored at a provider's mail server and *fetched* by the recipients using protocols like the Post Office Protocol POP3 [15] or the Internet Mail Access Protocol IMAP [16].

2.1 Protocol

Like most popular Internet services' protocols (except FTP), SMTP uses a single TCP connection to transfer both control commands and the actual mail data. A basic SMTP message sequence is sketched in Figure 2.

If the IP address of the mail server is not known, it is looked up using the Domain Name System (DNS). A TCP connection is then set-up to the mail server and the client waits for a greeting message from the SMTP server containing a `220` reply code and some greeting text. In a good-case scenario, the client will then send at least three commands which are all confirmed by the server with `250 (OK)` reply codes. The client greets the server using the `HELO` (or `EHLO` for ESMTP) command. The `MAIL FROM` command indicates the "envelope" mail return address and the `RCPT TO` command gives the "envelope" mail address of a recipient. The last command can be repeated if the mail is to be delivered to multiple recipients. When all commands have been acknowledged by the SMTP server, the client sends a `DATA` command and waits for the server's `354` reply, confirming the change of communication modes inside the TCP connection from command exchange to data transfer. The `DATA` command consists of six octets in a single packet (the string "`D A T A`" plus one linefeed and one carriage return character) and can thus be easily distinguished from the previous commands by its packet size.

The client then sends the mail message. Note that the popular RFC 822 mail headers (To:, From:, Cc:, Subject:, etc) are not transmitted in SMTP commands but are part of the mail message from the point of view of the SMTP

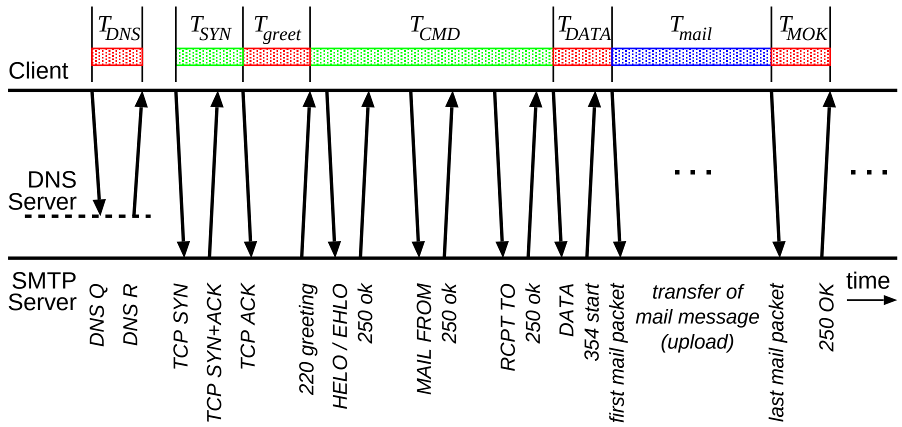

Figure 2: Message sequence for DNS lookup, TCP connection set-up, SMTP command exchange and mail upload. TCP Acknowledgment packets have been omitted after connection set-up.

communication. The address information in the headers can therefore differ from what was transmitted in the SMTP commands. The end of the mail message is indicated by the client sending a special string (a "." on a separate line), to which the server should respond with another `250 OK` message. The corresponding latencies for the different phases are defined in Figure 2.

2.2 Measurement Results

By analyzing Traces A, B and C, measurement results have been obtained for the different latencies as well as the number of commands preceding a mail upload and the sizes of the single mails. Their mean values and coefficients of variation[2] are summarized in Table 2 and visualized as associated bar graphs in Figure 3. A note of caution is necessary concerning the use of these mean values: The extremely high variance and the power tail property of most underlying distributions makes mean values neither stable nor very representative for the measured samples. Distributions will be given on double logarithmic scales below to allow extracting quantiles in addition to the measured mean and variance values given in Table 2.

A DNS lookup was observed before 40 % (23 % / 4 %) of all SMTP/TCP connections in Trace A (B/C). The lower rate of DNS requests in Trace C is due to the different measurement location at the access point of a large campus network where the traffic of clients asking a local DNS server was not captured and could therefore not be related to the start of an SMTP/TCP connection. Only computers that requested name resolution directly from remote DNS servers before opening an SMTP connection could be counted for the DNS statistics in Trace C. A brief look at the bars in Figure 3 indicates that the performance of e-mail uploads in high speed connections is severely limited by the command dialogues and server reaction times in SMTP. Only transfers of large e-mails can profit from high bandwidth connections.

[2]For a non-negative variable, the coefficient of variation is defined as the standard deviation divided by the mean value.

Table 2: Mean and coefficient of variation of latencies for the different stages of an SMTP connection, mail sizes and number of commands before a transfer. Random variables T_x denote latencies as defined in Figure 2. N_{CMD} is the number of commands preceding a mail transfer and S_{mail} the size of a mail message. c_v is the coefficient of variation[2].

	Trace A		Trace B		Trace C	
	mean	c_v	mean	c_v	mean	c_v
T_{DNS}	0.4 s	8.6	0.9 s	4.1	0.4 s	18.3
T_{SYN}	0.05 s	5.8	0.4 s	4.1	0.8 s	10.2
T_{greet}	0.4 s	8.1	0.6 s	6.4	5.8 s	2.5
T_{CMD}	1.1 s	2.3	2.0 s	3.3	2.5 s	5.8
N_{CMD}	4.2	0.3	4.4	0.4	4.3	0.4
T_{DATA}	0.08 s	0.6	0.5 s	2.9	1.5 s	13.3
T_{mail}	2.1 s	9.7	13.2 s	4.6	2.8 s	16.7
S_{mail}	88 kB	9.6	71 kB	4.9	33 kB	10.7
T_{MOK}	0.13 s	1.8	0.1 s	10.0	0.4 s	16.2

As most e-mails are first relayed to a local mail server even if they are addressed to far away places, the TCP connection set-up latency T_{SYN}, between the first client `SYN` and the client `ACK` packet acknowledging the server's `SYN+ACK` packet, is very low. The largest latency components are the SMTP commands dialogue and the mail transfer. The low speed access lines in Trace B lead to longer mail upload times than in Traces A and C.

As expected from previous investigations of Internet and Web traffic [17, 18], not only Web documents but also e-mail sizes have heavy-tailed distributions. A further look on the distribution of single e-mail sizes plotted as cumulative complementary distribution functions (CCDF) in Figure 4 reveals their heavy-tailed nature. A Pareto tail $\mathrm{P}\{S_{mail} > x\} = (x_0/x)^\alpha$ has been fitted to the distribution of mail sizes in Trace C in the range between 1 kB and 1 MB, with parameters $x_0 = 808\,B$ and $\alpha = 0.69$. Extrapolating this tail until infinity would not only yield an infinite variance (due to $\alpha \leq 2$) but also

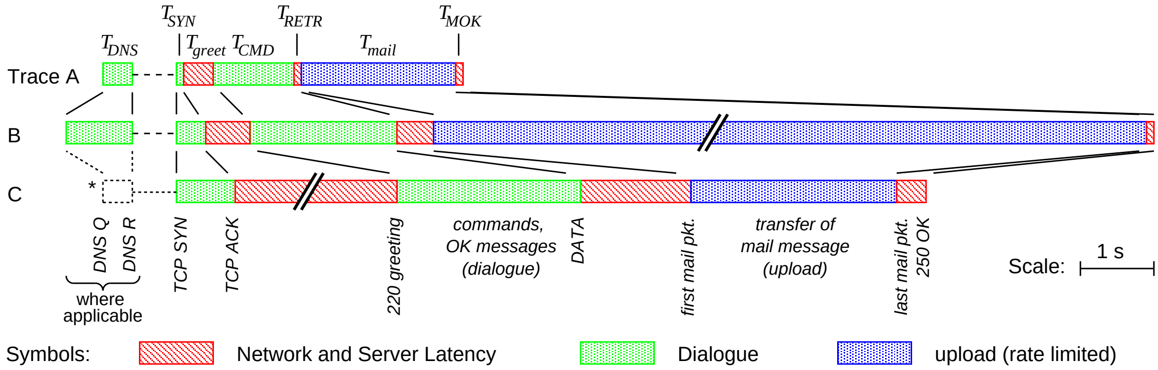

Figure 3: Mean latencies in SMTP connections for Traces A, B and C. In the server greeting time in Trace C a strong component due to a server waiting 30 s before greeting clients has been removed to improve readability. Only a part of the connections were preceded by DNS requests. *The DNS latencies for Trace C do not have the same statistical basis as in Traces A and B (see text).

an infinite expectation as $\alpha \leq 1$. Although half of the e-mails were less than 2 kB, around one in 100 had a size of more than 1 MB in all three Traces (including the modem/ISDN residential access!). Note that in practice, all measured distributions will show a finite maximum value. However, experience shows that measurements including more samples reveal larger maximum values and show distribution with power tails up to higher values.

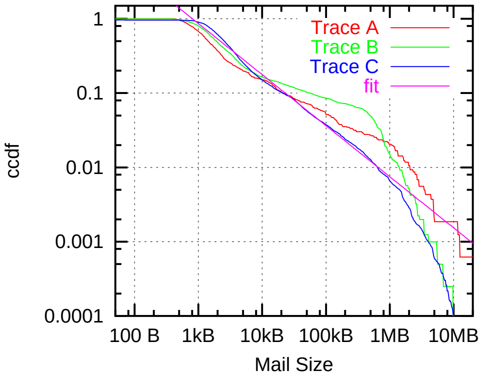

Figure 4: CCDF of SMTP mail size S_{mail}. Power tail fit included for Trace C with parameters $x_0 = 808\,B$ and $\alpha = 0.69$.

The high variance of e-mail file sizes S_{mail} is reflected in the duration T_{mail} of mail transfers, as can be seen in the variances given in Table 2 as well as in the CCDF plotted in Figure 5, which shows that although 90 % of all mail transfers can be done in under one second in the high speed scenarios of Traces A and C, one in 100 mails takes more than one minute to transfer due to its large size.

Whereas the high variance of e-mail sizes and the corresponding transfer times could be anticipated from the observation that file sizes tend to have heavy-tailed distributions, the fact that some other components of SMTP delay are also highly variant may be unexpected.

The server reaction times T_{greet} and T_{MOK} and the client and server round-trip reaction time T_{DATA} show a very high variance at least in Traces B and C. The

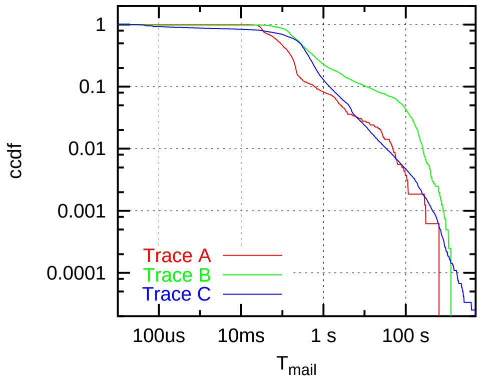

Figure 5: CCDF of SMTP mail upload latency T_{mail}.

server greeting time, depicted in Figure 6, additionally reveals a protocol problem in Trace C: In 13 % of all SMTP connections a mail server waited exactly 30 s after the TCP connection was established before greeting the client. This systematic defect may have several reasons. A failure of a reverse name look-up by the SMTP server or a failure to contact the IDENT service on the mail client may lead to a time-out, or the long latency is the result of overload protection on the mail server. Without these 13 % of the samples, the mean greeting latency in Trace C would be reduced from 5.8 s (Table 2) to 2.2 s, but the variance in this reaction time, ranging from around 100 ms up to more than one minute, would still be high.

The server reaction time for sending the `250 OK` reply after completion of the mail transfer is T_{MOK}. Its distribution is depicted in Figure 7. The power tail fit added for Trace C has parameters $x_0 = 75$ ms and $\alpha = 1.2$. Although in some cases an SMTP server can obviously send this application layer acknowledgment after less than 100 ms, there are cases when this process takes more than one minute. Message delivery problems related to excessive values of T_{MOK} have been reported as early as 14 years ago [19], but obviously not all recommendations of this RFC have been implemented yet.

As sketched in Figure 2, four commands and replies are

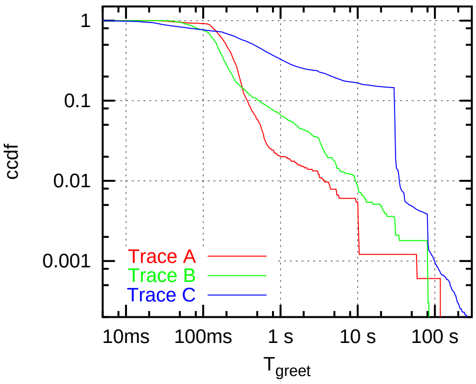

Figure 6: CCDF of SMTP greeting latency T_{greet}.

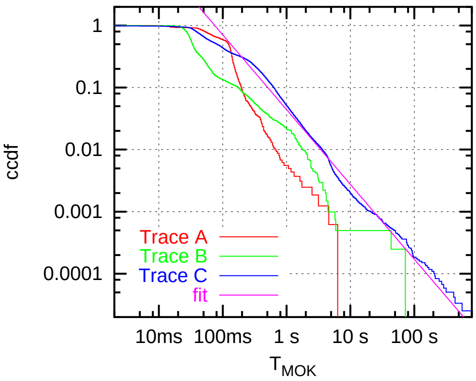

Figure 7: CCDF of the server reaction time T_{MOK} to acknowledge the complete transfer of an e-mail in an SMTP connection for Traces A, B and C. Power tail fit included for Trace C with parameters $x_0 = 75$ ms and $\alpha = 1.2$.

exchanged between SMTP client and server before the upload of an e-mail in the minimum "good" case at the beginning of an SMTP/TCP connection. To transfer further mails in the same connection, the client issues a reset command (`RSET`) and continues with the sequence of `MAIL FROM`, `RCPT TO` and `DATA`. Consequently, also later transfers in a connection are preceded by at least four commands. The distribution of N_{CMD} given in Figure 8 shows that, depending on the number of recipients of the message, this number can also be much higher. The concentration of probability at around 53 commands in Trace C (the final step in Figure 8 corresponds to more than 160 of the 324 k mail messages) is an indication of a popular limit of SMTP servers to accept mails for at most 50 recipients at the same time. Due to the much smaller number of mails in Traces A and B, the values in this probability range have no significance for those traces.

The distribution of the time T_{CMD} it takes to transmit and process the SMTP command/reply sequences is plotted in Figure 9. Although the number of commands before an e-mail is just four in most cases, a power law distribution is found between around 1 s and 100 s in Traces B and C for T_{CMD}. The Pareto tail fit added for Trace C has parameters $x_0 = 455$ ms and $\alpha = 1.12$. In addition to the variance in N_{CMD}, packet losses and retransmis-

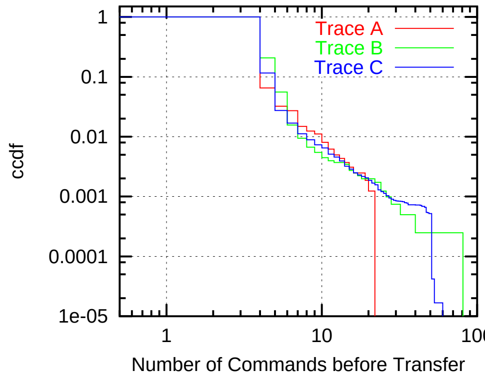

Figure 8: CCDF of the number of commands before an e-mail is uploaded via SMTP for Traces A, B and C.

sions with TCP's exponential back-off may be responsible for the high variance in T_{CMD}. This presumption is backed by the observation of around 1 % repeated client commands and around 3 % repeated server responses in Trace C, indicating packets lost on both sides behind the measurement point.

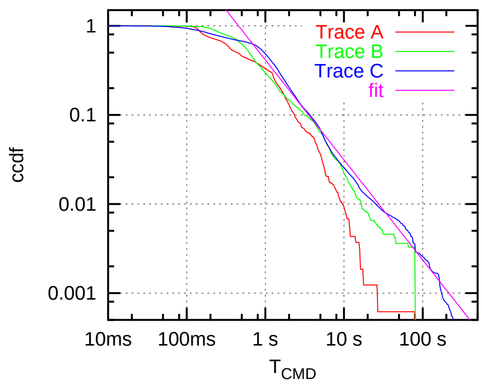

Figure 9: CCDF of duration of SMTP command exchange T_{CMD}. Power tail fit included for Trace C with parameters $x_0 = 455$ ms and $\alpha = 1.12$.

Whereas all of the above measures described characteristics or the performance related to single e-mails, the duration of an SMTP/TCP connection is also determined by the number of e-mails uploaded in the same connection. The distribution of the number of mails per connection is given in Figure 10. There is an obvious difference between the always-on scenarios of Traces A and C and the dial-up scenario of Trace B where the probability of transmitting multiple messages in an SMTP connection is one to two orders of magnitude higher, presumably because users tend to compose a number of e-mails before they dial into the network and then transmit them as a batch to the same provider's mail server.

3 Retrieving E-Mails (POP3/IMAP)

The Post Office Protocol, version 3 (POP3) [15] is usually employed for computers which are not always con-

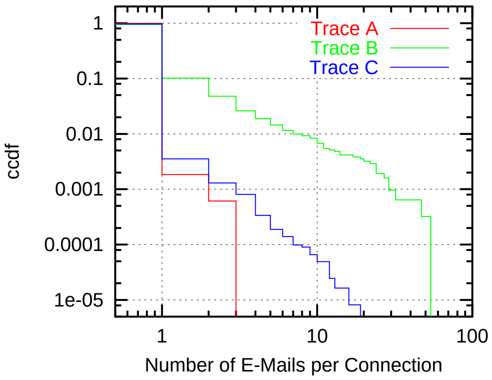

Figure 10: CCDF of the number of e-mails in an SMTP connection for Traces A, B and C.

nected to the Internet to access mailboxes on mail servers and retrieve mail messages to the local computer. An alternative protocol is the Internet Message Access Protocol (IMAP) [16], which is not as popular as POP3 but also allows remote mailbox administration. As there were only around 100 IMAP connections captured in Traces A and B, statistically sound results for IMAP can only be given for the analysis of Trace C.

3.1 Protocols

Similar to SMTP, the POP3 and IMAP protocols consist of server greeting, command exchange and mail transfer phases. In contrast to SMTP, POP3 and IMAP include user authentification after the server greeting, and the command exchange phases can be very short, as the most relevant information for the client (the number of e-mails in the mailbox) is sent by the server after authentication (and mailbox selection in the case of IMAP).

The sketch in Figure 11 shows the command sequence and the corresponding random variables for a message retrieval in POP3.

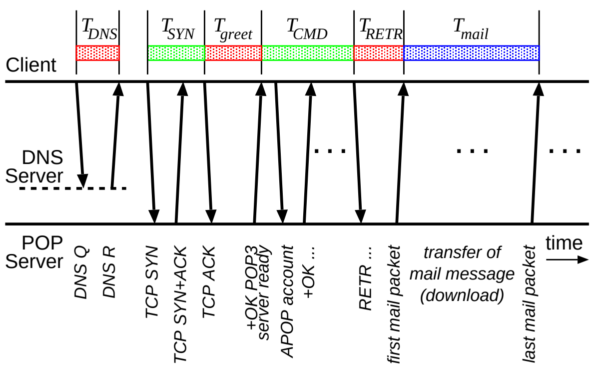

Figure 11: Message sequence for DNS lookup, TCP connection set-up, POP3 command exchange and mail download. TCP Acknowledgment packets have been omitted after connection set-up.

3.2 Measurement Results

The latencies, sizes of e-mails and numbers of e-mails transmitted per POP3 or IMAP connection observed in Traces A, B and C are summarized in terms of mean values and coefficients of variation in Table 3. The mean values from the POP3 results for the three traces are also visualized in Figure 12. The results for IMAP connections in Trace C have been omitted from Figure 12 as the extremely long mean duration of command exchanges did not fit in with the optical comparison to the POP3 results. For better visualization, T_{RETR} and T_{mail}, which both have mean values of more than 10 s in Trace B, had to be depicted out of scale for Trace B.

DNS lookups were observed before 11 % (16 %) of the POP3 connections captured in Trace A (B). As with SMTP, the location of the measurement point for Trace C restricted the DNS-POP3 or DNS-IMAP sequences recognized to those where the client host sent DNS requests directly to servers outside the local campus network. Correspondingly, the frequency of seeing a DNS lookup related to a POP3 (IMAP) connection in Trace C was only 0.2 % (2.3 %). As with SMTP (see Table 2), the mean DNS lookup latencies were below 1 s in most cases, but the observed variance was very high. The corresponding distribution functions for the comparable case of DNS lookups preceding HTTP/TCP connections can be found in [2] for Traces A and B.

The main component of mean delays in all traces is the command dialogue between client and server, even though it is impossible for the packet header analysis to distinguish between the retrieval of a short mail message and the mail server's output to a `LIST` command, listing the contents of a large mailbox (cf. Section 1.3.3). Consequently, the observed e-mail sizes are the result of a superposition of the actual distribution of e-mail sizes and the sizes of server replies to `LIST` commands. From the point of view of traffic modeling, this is not an issue as the corresponding traffic needs to be carried over the network. However, the size statistics for received mails are strongly influenced by the frequency of `LIST` commands, as can be seen in the plot of the CCDF of e-mail sizes in Figure 13 and in the mean values in Table 3.

A comparison of e-mail sizes in CFigures 4 and 13 shows that in Traces A and C (IMAP) the share of small transfers (multi-packet mailbox listings or very short e-mails) is high, leading to the mean message size being lower than in the SMTP case, which partially explains why the mean received message is smaller than the mean transmitted message. Besides this effect, the usual heavy-tailed distribution of file sizes can be seen nicely in the POP3 results of Trace C, where a power tail fit with parameters $x_0 = 0.53$ kB and $\alpha = 0.83$ has been added to the graph, closely following the distribution of message sizes between 1 kB and 1 MB.

The greeting latency T_{greet} observed from POP3 and IMAP servers was shorter than in the SMTP case, as a comparison of CFigures 6 and 14 reveals where espe-

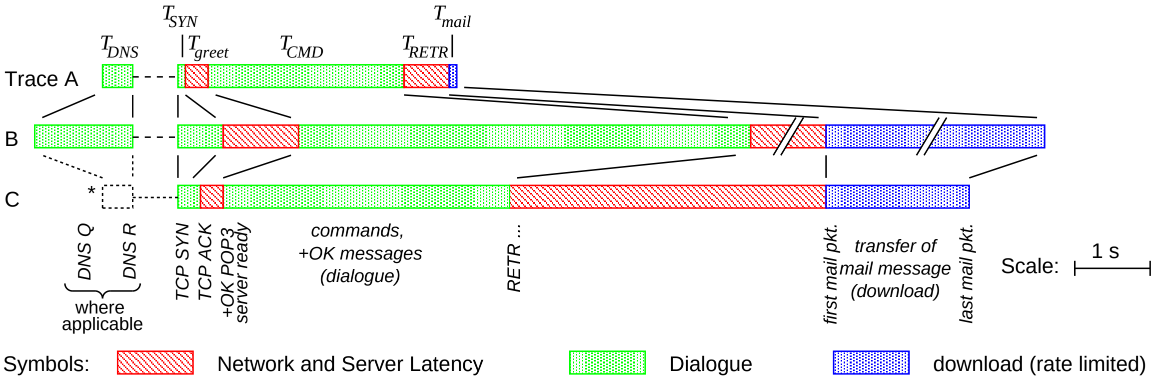

Figure 12: Mean latencies in POP3 connections for Traces A, B and C. In order to maintain readability, the extremely large mean values of T_{RETR} and T_{mail} have not been drawn to scale for Trace B. Only a part of the connections were preceded by DNS requests. *The DNS latencies for Trace C do not have the same statistical basis as in Traces A and B (see text).

Table 3: Mean and coefficient of variation of latencies for the different stages of a POP3 or IMAP connection, mail sizes S_{mail} and number of e-mails N_{mails} retrieved during one connection. Random variables T_x denote the latencies defined in Figure 11.

	Trace A: POP3		Trace B: POP3		Trace C: POP3		Trace C: IMAP	
	mean	c_v	mean	c_v	mean	c_v	mean	c_v
T_{DNS}	0.4 s	8.6	0.9 s	4.1	0.4 s	18.3	5.7 s	12.7
T_{SYN}	0.06 s	15.0	0.6 s	2.7	0.3 s	27.1	0.8 s	1.4
T_{greet}	0.3 s	11.7	1.1 s	2.2	0.3 s	4.5	0.4 s	1.3
T_{CMD}	2.6 s	1.6	5.0 s	5.3	3.8 s	4.2	42.0 s	1.1
T_{RETR}	0.6 s	5.0	94 s	4.6	4.1 s	3.0	0.2 s	1.5
T_{mail}	0.1 s	47.2	10.4 s	8.4	1.9 s	17.9	0.2 s	1.8
S_{mail}	4.2 kB	12.7	36.0 kB	8.2	10.4 kB	10.9	2.8 kB	0.8
N_{mails}	0.15	8.8	0.41	6.6	0.06	11.3	1.18	2.1

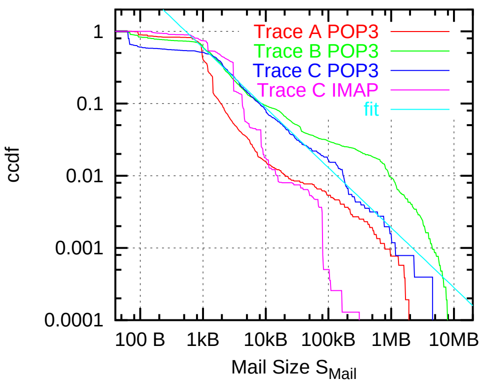

Figure 13: CCDF of POP3 and IMAP mail sizes. Power tail fit included for Trace C (POP3) with parameters $x_0 = 0.53$ kB and $\alpha = 0.83$.

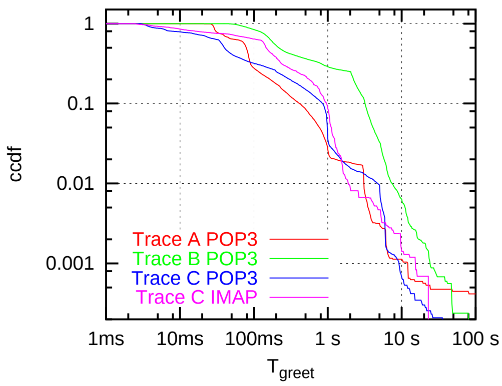

Figure 14: CCDF of observed greeting latency T_{greet} of POP3 or IMAP servers.

cially the tail probabilities are around one order of magnitude lower for POP3 and IMAP servers than for SMTP servers.

The POP3 or IMAP command exchange phase in all traces have been observed to take a very long time in some cases. The longest exchanges were observed with IMAP, due to the slightly different mode of operation compared to POP3: An IMAP client can maintain an open connection to the IMAP server and – instead of repeatedly polling the server as a POP3 client would do – wait until the server notifies the client of new e-mails. In the trace analysis, this is counted as part of the command exchange phase, leading to the observed exceptionally high values of T_{CMD} for IMAP. Also counted in T_{CMD} are the cases when POP3 clients leave connections open to the server and repeatedly poll the server to see if new messages have arrived. This behavior seems to have found increased popularity since the time when Trace A was recorded, as the distribution tails for T_{CMD} found in Traces B and C differ significantly from Trace A.

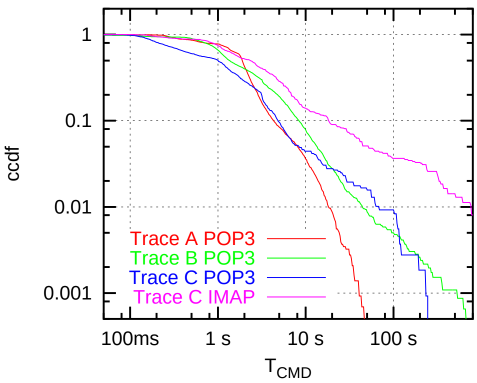

Figure 15: CCDF of duration of POP3 or IMAP command exchanges T_{CMD}.

The time it took (after a POP3 `RETR` or an IMAP `FETCH` command) to actually download an e-mail was found to vary over more than six orders of magnitude. Whereas 86 % of the retrievals (mostly multi-packet mailbox listings) in Trace A took less than around 1 ms, the actual transfer of large e-mails could easily take more than a minute. Especially the transfer of large e-mails over the slower modem or ISDN lines in Trace B were observed to take more than 15 min.

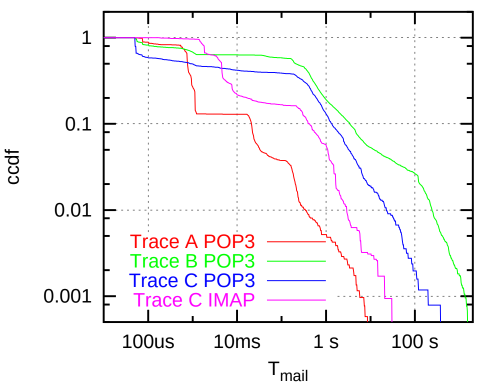

Figure 16: CCDF of POP3 or IMAP mail download durations T_{mail}.

The distributions of the number of e-mails retrieved in the same POP3 or IMAP connection are depicted in Figure 17. The intersection points of the complementary distributions with the vertical axis give the ratio of connections transporting at least one message: Only in 12 % (Trace A), 18 % (Trace B), 2.5 % (Trace C:POP3) and 25 % (Trace C:IMAP) of all connections there was at least one e-mail transferred. The rest of the connections were used only to check for new mails, with the server telling the client that there were no new mails available. The number of POP3 connections where new mails were downloaded is higher in Trace B than in Traces A and C because users with a time-charged dial-up connection to the Internet tend to check their mailboxes less frequently than "always-on" users.

On the other hand, the share of connections with e-mail transfers is higher with IMAP than with POP3. An IMAP server can automatically notify a mail client within an existing IMAP/TCP connection when a new e-mail arrives. Therefore it is advantageous for the client to keep an IMAP connection open to the mail server and wait for new mails even if initially there was no new mail available. Consequently, the mean duration of IMAP connections (388 s in Trace C) was around 50 times longer than that of POP3 connections (8 s in Trace C and 7 s in Trace A) and the mean number of messages in an IMAP connection was also found to be much larger than in a POP3 connection (see Table 3).

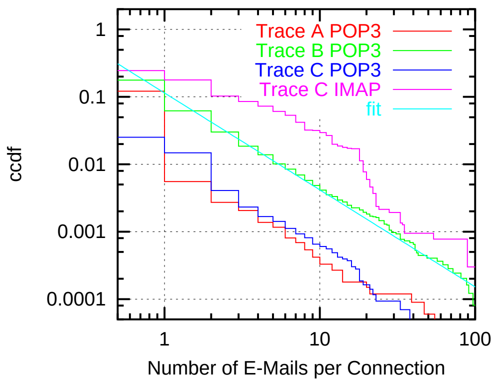

Figure 17: CCDF of the number of E-Mails retrieved in a POP3 or IMAP connection. Power tail fit included for Trace C with parameters $x_0 = 0.22$ and $\alpha = 1.44$.

Although the majority of the connections did not transport any e-mail message at all, some connections were used to download a large number of e-mails. The graphs in Figure 17 indicate that the number of e-mails found in POP3 connections follows a power tail distribution, although the data only support this statement for two orders of magnitude. The power tail fit added to the graph from Trace B has parameters $x_0 = 0.22$ and $\alpha = 1.44$. As mentioned above, the dial-up users observed in Trace B polled their mailboxes less frequently. Consequently, the number of messages retrieved per connection in Trace B was correspondingly higher. The distribution has the same general shape as that of N_{mails} in Traces A or C but its tail shows values one order of magnitude higher in probability.

With an average duration of 23.5 s, the POP3 connections observed in Trace B lasted three times as long as those observed in Traces A and C. On the one hand, this is due to the lower access line speed of modem/ISDN lines compared to LAN or ADSL access to the Internet, which causes the download of an e-mail message to take longer. On the other hand, the inconvenience and cost associated with the dial-in procedure prevents users from checking their mail more frequently. Consequently, the chance of retrieving a new e-mail message in a POP3 connection and the number of messages retrieved in each connection is increased.

4 Conclusions

Exchanging e-mail messages is – after Web access and file sharing – one of the most popular and widely used service offered on the Internet. Although TCP makes e-mail traffic elastic in terms of the required data rate and although e-mail is one of the classic "store and forward" technologies (if regarded on the application level), users expect interactive performance not necessarily for the end-to-end delivery of e-mail messages but for their interaction with a mail program's user interface. E-mail programs that block access to a local mailbox or even to the whole computer while e-mails are transmitted to or retrieved from a mail server have shifted this traditionally asynchronous service into the focus of Internet performance analysis.

Extensive application-level analysis of packet header traces allowed us to evaluate traffic characteristics and performance measures for SMTP, POP3 and IMAP connections as they occurred in real life when the traces were recorded. For the majority of small transfers, the performance of all protocols is severely limited by dialogues between client and server and server reaction times, which cannot be significantly reduced by increasing link bandwidth on the Internet but calls for operators to offer low latency access technologies.

It was expected and confirmed that e-mails have the same kind of highly variant size distributions as most files transferred on the Internet. In addition, high variance distributions with power tail exponents $\alpha = 0.7 \ldots 1.5$ were also found in other measures like the number of e-mails retrieved in a POP3 connection and in most reaction times or dialogue durations.

Acknowledgments

The author would like to thank the people at Deutsche Telekom, University of Münster and Bürgernetz Fünfseenland who allowed him to record the traffic traces A and B used in this paper. Special thanks go to the WAND and NLANR MOAT research groups for making their traces available online.

References

[1] J.B. Postel, "Simple Mail Transfer Protocol," RFC 821, 1982.

[2] J. Charzinski, "Web Performance in Practice – Why We are Waiting," *AEÜ Intl. J. Elec. Comm.*, vol. 55, pp. 37–45, 2001.

[3] J. Charzinski, "Measured HTTP Performance and Fun Factors," in *Proc. ITC*, Salvador, BA, Brasil, 2001, pp. 1063–1074.

[4] C. Cáceres, P.B. Danzig, S. Jamin, and D.J. Mitzel, "Characteristics of Wide-Area TCP/IP Conversations," in *Proc. ACM Sigcomm*, Zurich, Switzerland, 1991, pp. 1991–112.

[5] F.D. Smith, F. Hernandez Campos, K. Jeffay, and D. Ott, "What TCP/IP Protocol Headers Can Tell Us About the Web," in *Proc. ACM Sigmetrics*, Cambridge, MA, USA, 2001, pp. 245–256.

[6] B. Krishnamurthy and C.E. Wills, "Analyzing factors that influence end-to-end Web performance," *Comp. Netw.*, vol. 33, pp. 17–32, 2000.

[7] P. Barford and M.E. Crovella, "Measuring Web Performance in the Wide Area," *ACM Perf. Eval. Review*, vol. 27, no. 2, pp. 37–48, 1999.

[8] A. Habib and M. Abrams, "Analysis of Sources of Latency in Downloading Web Pages," in *Proc. Webnet*, San Antonio, TX, USA, 2000.

[9] E. Cohen and H. Kaplan, "Prefetching the Means for Document Transfer: A New Approach for Reducing Web Latency," in *Proc. IEEE Infocom*, Tel Aviv, Israel, 2000.

[10] C. Huitema and S. Weerahandi, "Internet Measurements: the Rising Tide and the DNS Snag," in *Proc. ITC Spec. Sem. on IP Traffic*, Monterey, CA, USA, 2000, pp. 2.1–2.10.

[11] J. Charzinski, "HTTP/TCP Connection and Flow Characteristics," *Perf. Eval.*, vol. 42, pp. 149–162, 2000.

[12] WAND Research Group, "Auckland-IV Trace," http://pma.nlanr.net/Traces/long/auck4.html, accessed Apr. 2002.

[13] "WAND Research Group at University of Waikato," http://wand.cs.waikato.ac.nz/, accessed Apr. 2002.

[14] "NLANR Passive Measurements and Analysis Research Group," http://moat.nlanr.net/PMA/, accessed Apr. 2002.

[15] J. Myers and M. Rose, "Post Office Protocol – Version 3," RFC 1939, 1996.

[16] M. Crispin, "Internet Message Access Protocol – Version 4rev1," RFC 2060, 1996.

[17] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson, "On the Self-Similar Nature of Ethernet Traffic," in *Proc. ACM Sigcomm*, San Francisco, CA, USA, 1993, pp. 183–193.

[18] M.E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Trans. Networking*, vol. 5, pp. 835–846, 1997.

[19] C. Partridge, "Duplicate Messages and SMTP," RFC 1047, 1988.