

Internet e-mail



protocols,



standards, and

implementation



CD-ROM
included

Lawrence Hughes

**Internet E-mail:
Protocols, Standards, and
Implementation**

Internet E-mail: Protocols, Standards, and Implementation

Lawrence Hughes



Artech House
Boston • London

Library of Congress Cataloging-in-Publication Data

Hughes, Lawrence E.

Internet E-mail : protocols, standards, and implementation /
Lawrence Hughes.

p. cm. — (Artech House telecommunications library)

Includes bibliographical references and index.

ISBN 0-89006-939-5 (alk. paper)

1. Electronic mail systems. 2. Internet (Computer network)

I. Title. II. Series

TK5105.73.H84 1998

004.692—dc21

98-20082

CIP

British Library Cataloguing in Publication Data

Hughes, Lawrence

Internet E-mail : protocols, standards, and implementation.

—(Artech House telecommunications library)

1. Electronic mail systems 2. Internet (Computer network)

I. Title

004.6'92

ISBN 0-89006-939-5

Cover and text design by Darrell Judd

© 1998 ARTECH HOUSE, INC.

685 Canton Street

Norwood, MA 02062

All rights reserved. Printed and bound in the United States of America. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Artech House cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

International Standard Book Number: 0-89006-939-5

Library of Congress Catalog Card Number: 98-20082

Oracle Exhibit 1013

Page 6 of 15

*To Dylan Edwin Hughes, who was born during the writing of this book.
My wife is from the Philippines, and she returned there for Dylan's birth.
My first glimpse of my new son was via a scanned-in JPEG image attached
to an e-mail message and sent halfway around the world in a matter of
seconds. Dylan will grow up in a world of instant global communication
that I can scarcely imagine. I hope this book will contribute in some small
way to making his world the best possible one.*

that user on $MSS[j]$ via an internal IPC connection between $MTA[i]$ and $MSS[j]$.

The basic message flow (as seen in Figure 2.2) for a POP-based retrieving messages from a distributed mail server is as follows. An IMAP-based one would work exactly the same way but use the IMAP protocol in place of the POP protocol and port 143 in place of port 110.

1. The UA requests the DNS to resolve the shared nodename for the POP server (e.g., pophost.megacorp.com), and the DNS responds with one of the m IP addresses corresponding to the m POP servers in the system (it does not matter which one—call it $POP[k]$).
2. The UA connects to $POP[k]$ on TCP port 110.
3. $POP[k]$ uses the Routing Table to determine whether the specified user account is a valid one, and if so, which MSS contains messages for that user (call it $MSS[j]$).
4. Assuming it is a valid user, $POP[k]$ makes an internal IPC connection to $MSS[j]$ and allows the UA to retrieve messages from that server via the POP protocol.

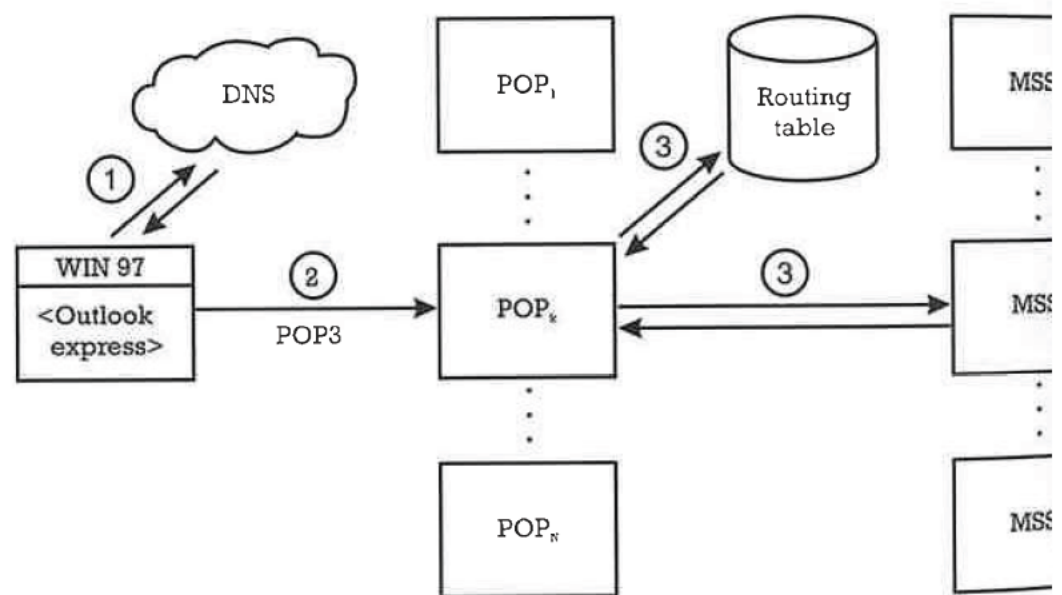


Figure 2.2 Retrieving message from Scalable E-mail Server with POP3.

Basic message exchange functionality

The core functionality of Internet e-mail is concerned with sending and retrieving messages. The overall design is known as a store and forward system. All that means is that the sender and the receiver need not be involved in the message exchange at the same time, as they would with simple voice telephones. It is more like leaving messages for each other on answering machines. However, if a telephone answering machine records an incoming message for someone other than the machine's owner, it generally is not clever enough to realize that and forward it to the correct answering machine (Internet MTAs do exactly that).

We will describe two scenarios involving sending and receiving Internet Mail messages. The first one is the simpler one, because no inter-MTA forwarding is required, while the second one does involve such forwarding.

Scenario 1: sender and receiver on the same postoffice

Say we have a TCP/IP domain called guys.com, with an Internet Mail postoffice (both SMTP and POP3 servers) on node *mail* in that domain, as shown in Figure 2.3. Furthermore, we have two guys named Albert and Barney who have Internet-style UAs on their computers. Both of them specified the nodename mail.guys.com for both smtp host and pop3 host when they configured their UAs. Say that Albert wants to send mail to Barney. The following steps must take place:

- S1. Albert starts his UA and composes a new message addressed to Barney (he could even use the more formal style Barney@guys.com, but within a single postoffice that is not necessary). He then clicks his Send button.
- S2. In response to the Send button being clicked, Albert's UA makes a network (TCP) connection to the node mail.guys.com on port 25, which is the port used by the SMTP server. The SMTP server there accepts the connection.

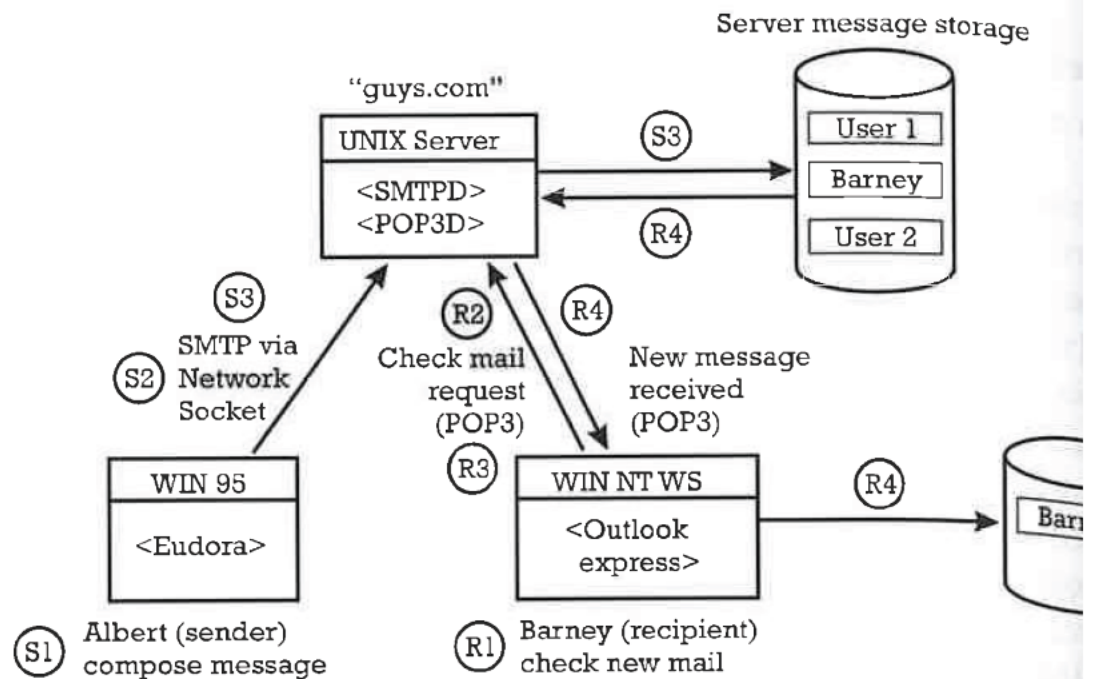


Figure 2.3 Scenario 1: Internet e-mail system: sender and recipient on same server.

S3. As soon as the connection is established, Albert's UA identifies Albert as the sender and Barney as the recipient (which are acceptable to the SMTP server). Albert's UA then sends the message (including the message headers, the message body, the "signature," if any). Once the message is received, the SMTP server looks at the recipient, realizes that the mail is intended for someone who has a local account in this postoffice (Barney), and hands it to the MS to file in that user's mailbox.

S4. Albert's UA terminates the network connection.

At some point, Barney will complete the mail exchange by retrieving the message Albert just sent. The following steps take place:

R1. Barney starts his UA and clicks the Check for New Mail button (perhaps his UA has been running all the time and checking for new mail every few minutes).

R2. In response to the Check for New Mail button being clicked, Barney's UA makes a TCP network connection to the

mail.guys.com on port 110, which is the port used by the POP3 server. The POP3 server accepts the connection.

R3. As soon as the connection is established, Barney's UA identifies him as the user and supplies his POP3 password. Barney has a valid POP3 account and specified the correct password for that account, so all this is accepted also.

R4. Barney's UA requests a list of messages in his area of the MS and realizes a new one is there. The UA then retrieves the entire message over the network connection and stores it in Barney's local MS (in his InBox folder). Barney did not select the Leave Messages on Server option in his UA, so his UA tells the POP3 server to delete the new message, which it dutifully marks for deletion.

R5. Barney's UA realizes there are not any other new messages, so it sends a Quit message, which causes the POP3 server to actually delete the marked message, and then terminates the network connection.

R6. A summary line for the new message magically appears in Barney's InBox folder, showing that it was sent by Albert. Barney can view the message, print it, reply to it, forward it on to Charley, delete it, or do whatever he wants with it.

Scenario 2: sender and receiver on different postoffices

Say we have the same TCP/IP domain called guys.com, with an Internet Mail postoffice (both SMTP and POP3 servers) on node mail. Say we also have another TCP/IP domain elsewhere on the Internet, called gals.com. This is shown in Figure 2.4. We have good old Albert, still in the domain guys.com, and Alice in gals.com, both with UAs on their respective computers. Remember that Albert specified his SMTP and POP3 hosts both as mail.guys.com. However, in the gals.com domain, the SMTP server is on node smtp-mail and the POP3 server is on a different server called pop_mail. So Alice specified smtp_mail.gals.com for her SMTP host and pop_mail.gals.com for her POP3 host. Say that Albert wants to send mail to Alice. The following steps must take place:

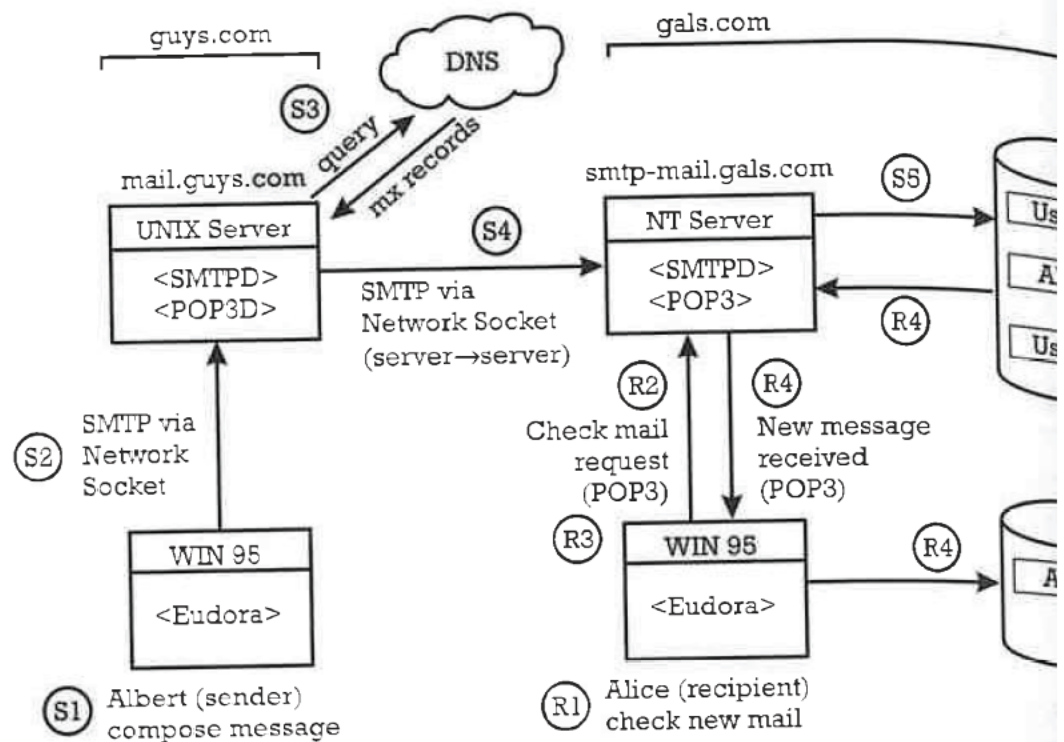


Figure 2.4 Scenario 2: Internet e-mail system sender and recipient on different servers.

- S1. Albert starts his UA and composes a new message addressed to Alice@gals.com (he cannot use the short form "Alice" because Alice does not have an account on the local postoffice). He clicks his Send button.
- S2. In response to the Send button being clicked, Albert's UA makes a TCP network connection to the node mail.guys.com on port 25, which is the port used by the SMTP server. His SMTP server (mail.guys.com) accepts the connection.
- S3. Albert's UA identifies him as the sender and Alice@gals.com as the recipient (which are both acceptable to the server). His UA sends the actual message. His SMTP server realizes that this mail is addressed to someone in another domain. It does not store it in its local MS. Instead, it does a query on the DNS server to see what the preferred node is to accept mail for the domain gals.com. If mail has been sent recently from guys.com to gals.com, his local DNS may have the answer right off the top of its "head" (in cache). If not, his DNS will ask other DNS servers for the answer.

until one (possibly the one for gals.com) admits to having that information and passes it back to Albert's DNS server. Eventually, the answer returns that smtp-mail.gals.com is the preferred node for incoming mail for gals.com.

- S4. On finding the name of the node that handles incoming mail for the domain gals.com, Albert's SMTP server establishes a TCP network connection (possibly via WAN links or even halfway around the world) to the SMTP server on that node (smtp-mail.gals.com). To that SMTP server, the connection looks just like one from a local client and is handled in exactly the same way.
- S5. Albert's SMTP server identifies Albert (of guys.com) as the sender and Alice (of gals.com) as the recipient (which are both acceptable to the SMTP server). Albert's SMTP server sends the actual message. Alice's SMTP server realizes that the mail is addressed to someone with a local account and hands it to the message store to file in Alice's mailbox.
- S6. Albert's SMTP server terminates the network connection.

At some point in the future, Alice will complete the mail exchange by retrieving the message Albert just sent. The following steps take place:

- R1. Alice starts her UA and clicks the Check for New Mail button (or perhaps her UA has been running all the time and checking for new mail every few minutes).
- R2. Alice's UA makes a connection to node pop-mail.gals.com on port 110, which is the port used by the POP3 server. The POP3 server accepts the connection.
- R3. Alice's UA identifies her as the user and supplies her POP3 password. Alice has a valid POP3 account and specified the correct password, so all this is accepted also.
- R4. Alice's UA then requests a list of messages in her mailbox on the MS and realizes that there is a new one there. It then retrieves the entire message over the network connection and stores it in Alice's local MS (in her InBox folder). Unlike Barney's UA, Alice's UA

select the Leave Messages on Server option in her UA, so he does not tell the POP3 server to delete the new message. However, her UA makes a note of the unique message identifier (UIDL) for that message, so it will not download the message again.

- R5. Alice's UA realizes there are not any other new messages, sends a Quit message and terminates the network connection.
- R6. A summary line for the new message magically appears in Alice's InBox folder, showing that it was sent by Albert, on guys.com. Alice can view the message, print it, reply to it, forward it on to Barbara, delete it, or do whatever she wants with it.

Basically all other kinds of mail transactions come down to one or the other of those scenarios, as far as the mail servers are concerned. All other functionality (e.g., replying, forwarding mail, etc.) is done by the UA (e.g., by swapping the contents of the From: and To: header lines in a reply).

It is possible to tell an Internet Mail MTA to always send outgoing mail to a single node rather than doing distribution directly to the destination nodes. That allows setting up gateway or proxy systems for more efficient routing. One reason for doing that might be to hide the internal structure of the e-mail servers in a large company from the outside world. Internal servers could be added or moved without the outside world needing to know about such changes. Another reason might be so that incoming mail would come through a single proxy server, which could be thought of as a carefully controlled hole in an otherwise impermeable firewall protecting your LAN from unauthorized outside snooping and malicious tampering. There are even anonymizing gateways through which you can route mail that will strip any trace of where the mail originally originated.

CHAPTER

3

Contents

Jargon

Attachments

uuencode/uudecode:
the UNIX attachment
standard

BINHEX: the "Mac"
attachment standard

MIME: the Internet
standard

Examples of using
attachments in e-mail

Address books

Generalized directory
services

Attachments and address books

The concepts of attachments and address books go beyond the basic functions of an e-mail system, but they are important to its usability. While both areas have had good solutions in proprietary mail systems for a number of years, there is considerable variation at present in the level of sophistication, interoperability, and even the existence of these features in current Internet e-mail products. In the case of attachments, an excellent standard (MIME) is already available and is being supported (at least in part) by more and more clients. In the case of address books, several standards exist for shared address books, but none has been widely adopted so far. However, a very good standard is well underway and may finally solve that difficult problem on a global scale.