

HEIG-VD DAI - SMTP and ncat

[Link to the course](#)

L. Delafontaine and H. Louis, with the help of [GitHub Copilot](#).

Based on the original course by O. Liechti and J. Ehrensberger.

This work is licensed under the [CC BY-SA 4.0](#) license.

Objectives

- Learn electronic messaging protocols:
 - SMTP
 - POP3
 - IMAP
- Focus on the SMTP protocol
- Learn how to use ncat and Java to send an email to an SMTP server



Electronic messaging protocols: SMTP, POP3 and IMAP

More details for this section in the [course material](#). You can find other resources and alternatives as well.

Electronic messaging protocols: SMTP, POP3 and IMAP

- Email clients are called **Mail User Agents (MUA)**
- Email servers are called **Mail Transfer Agents (MTA)**
- They use different protocols to communicate



SMTP

- SMTP: Simple Mail Transfer Protocol
- Uses TCP port 25 (unencrypted) or 465 (encrypted)
- Used to send emails

The screenshot shows the IETF Datatracker page for RFC 5321, titled "Simple Mail Transfer Protocol". The page is from the Network Working Group, authored by J. Klensin in October 2008. It is a Request for Comments (RFC) 5321, categorized as a Standards Track document. The page includes a "Status of This Memo" section, an "Abstract" section, and a "Document type" section. The "Document type" section indicates it is an RFC Draft Standard, updated by RFC 7504, obsoletes RFC 2821, and updates RFC 1123. It was drafted by John C. Klensin. The "Select version" section shows a table of versions, with RFC 5321 selected. The "Compare versions" section shows a dropdown menu for comparing versions, with RFC 5321 selected. The "Author" section lists Dr. John C. Klensin as the author.

Network Working Group
Request for Comments: 5321
Obsoletes: [2821](#)
Updates: [1123](#)
Category: Standards Track

J. Klensin
October 2008

Simple Mail Transfer Protocol

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This document is a specification of the basic protocol for Internet electronic mail transport. It consolidates, updates, and clarifies several previous documents, making all or parts of most of them obsolete. It covers the SMTP extension mechanisms and best practices for the contemporary Internet, but does not provide details about particular extensions. Although SMTP was designed as a mail transport and delivery protocol, this specification also contains information that is important to its use as a "mail submission" protocol for "split-UA" (User Agent) mail reading systems and mobile environments.

Document type
RFC Draft Standard
October 2008
[View errata](#)
[Report errata](#)
Updated by [RFC 7504](#)
Obsoletes [RFC 2821](#)
Updates [RFC 1123](#)
Was [draft-klensin-rfc2821bis](#) (individual in app area)

Select version

00	01	02	03	04
05	06	07	08	09
10	11	RFC 5321		

Compare versions
...nsin-rfc2821bis-10
RFC 5321
[Side-by-side](#) [Inline](#)

Author
[Dr. John C. Klensin](#) ✉
[Email author](#)

POP3

- POP3: Post Office Protocol
- Uses TCP port 110 (unencrypted) or 995 (encrypted)
- Used to retrieve emails from a server

The screenshot displays the IETF Datatracker page for RFC 1939, 'Post Office Protocol - Version 3'. The page is organized into several sections:

- Network Working Group:** Request for Comments: 1939, STD: 53, Obsoletes: [1725](#), Category: Standards Track.
- Authors:** J. Myers (Carnegie Mellon), M. Rose (Dover Beach Consulting, Inc.), May 1996.
- Status of this Memo:** This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.
- Table of Contents:**
 - 1. Introduction 2
 - 2. A Short Digression 2
 - 3. Basic Operation 3
 - 4. The AUTHORIZATION State 4
 - 5. The TRANSACTION State 5
 - 6. The UPDATE State 10
 - 7. Optional POP3 Commands 11
 - 8. Scaling and Operational Considerations 16
 - 9. POP3 Command Summary 18
 - 10. Example POP3 Session 19
 - 11. Message Format 19
 - 12. References 20
 - 13. Security Considerations 20
 - 14. Acknowledgements 20
 - 15. Authors' Addresses 21
 - Appendix A. Differences from [RFC 1725](#) 22
- Document type:** RFC, Internet Standard, May 1996.
- Select version:** 00, 01, 02, 03, 04, 05, 06, **RFC 1939**.
- Compare versions:** ...yers-pop-pop3-05, RFC 1939.
- Authors:** Dr. Marshall T. Rose, John G. Myers.

IMAP

- IMAP: Internet Message Access Protocol
- Uses TCP port 143 (unencrypted) or 993 (encrypted)
- Used to retrieve emails from a server
- Much more powerful than POP3 (synchronization, ...)

Network Working Group
Request for Comments: 3501
Obsoletes: [2060](#)
Category: Standards Track

M. Crispin
University of Washington
March 2003

INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

The Internet Message Access Protocol, Version 4rev1 (IMAP4rev1) allows a client to access and manipulate electronic mail messages on a server. IMAP4rev1 permits manipulation of mailboxes (remote message folders) in a way that is functionally equivalent to local folders. IMAP4rev1 also provides the capability for an offline client to resynchronize with the server.

IMAP4rev1 includes operations for creating, deleting, and renaming mailboxes, checking for new messages, permanently removing messages, setting and clearing flags, [RFC 2822](#) and [RFC 2045](#) parsing, searching, and selective fetching of message attributes, texts, and portions thereof. Messages in IMAP4rev1 are accessed by the use of numbers. These numbers are either message sequence numbers or unique identifiers.

IMAP4rev1 supports a single server. A mechanism for accessing configuration information to support multiple IMAP4rev1 servers is discussed in [RFC 2244](#).

IMAP4rev1 does not specify a means of posting mail; this function is handled by a mail transfer protocol such as [RFC 2821](#).

Document type
RFC **Proposed Standard**
March 2003

[View errata](#)
[Report errata](#)

Obsoleted by [RFC 9051](#)
Updated by [RFC 4466](#),
[RFC 4469](#), [RFC 4551](#),
[RFC 5032](#), [RFC 5182](#),
[RFC 5738](#), [RFC 6186](#),
[RFC 6858](#), [RFC 7817](#),
[RFC 8314](#), [RFC 8437](#),
[RFC 8474](#), [RFC 8996](#)
Obsoletes [RFC 2060](#)
Was [draft-crispin-imapv](#)
(individual in app area)

Select version

00	01	02	03	04
05	06	07	08	09
10	11	12	13	14
15	16	17	18	19
20	RFC 3501			

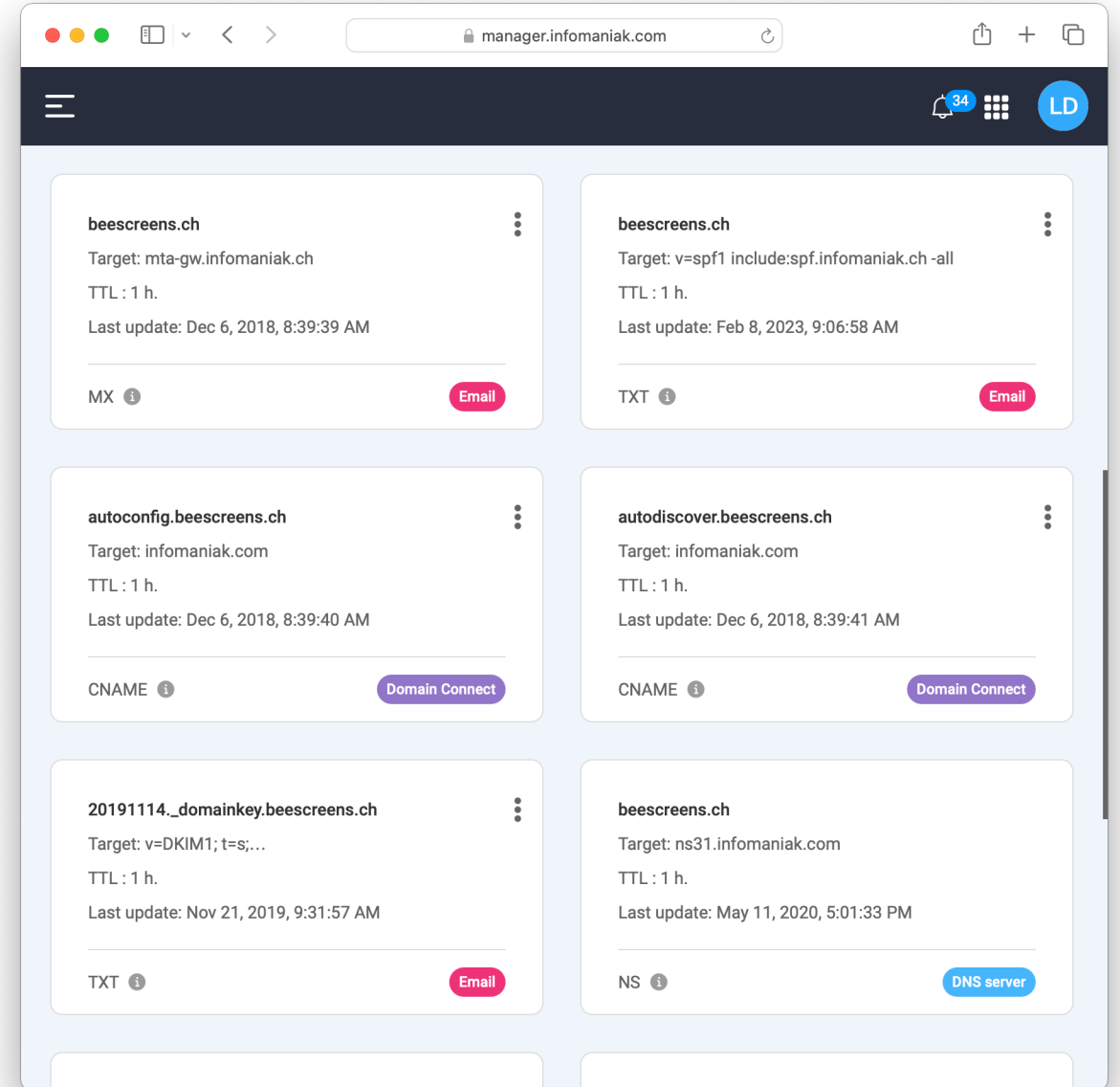
Compare versions

DNS records related to email

More details for this section in the [course material](#). You can find other resources and alternatives as well.

DNS records related to email

- **MX** : Mail eXchange - Specifies the mail server responsible for a domain name
- **TXT** : Store any text-based information. Used for **SPF** records, for email authentication

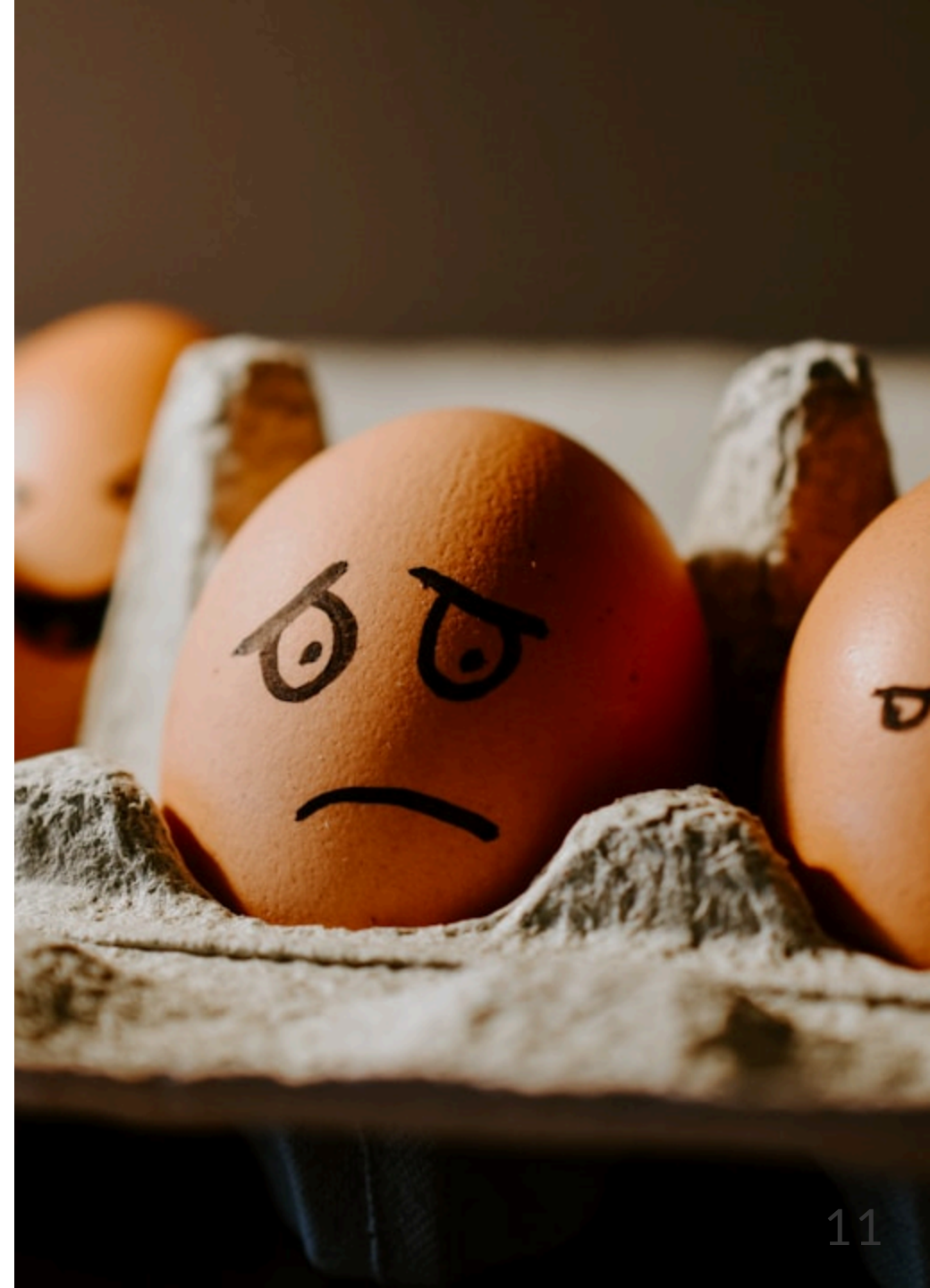


Security concerns and spam

More details for this section in the [course material](#). You can find other resources and alternatives as well.

Security concerns and spam

- SMTP is old and insecure
- Easy to spoof and forge emails
- Hard to maintain
- ▶ Your email server can be used for spam and can be blocked
- ▶ We will use a mock server to simulate an email server



A focus on the SMTP protocol

More details for this section in the [course material](#). You can find other resources and alternatives as well.

A focus on the SMTP protocol

- SMTP is a text-based protocol
- Messages are sent by the client to the server
- The server responds with a status code
- The client can send the next command

The screenshot shows the IETF Datatracker page for RFC 5321, titled "SMTP". The page is viewed in a web browser at datatracker.ietf.org. The main content area displays the protocol specification, including the "Attdl-Protocol" section and the "4.5. Additional Implementation Issues" section, which further details the "4.5.1. Minimum Implementation". The text specifies that SMTP is a text-based protocol where messages are sent from client to server, and the server responds with status codes. It also lists the minimum implementation commands: EHLO, HELO, MAIL, RCPT, DATA, RSET, NOOP, QUIT, and VRFY. The right sidebar provides metadata for the document, including its type (RFC Draft Standard), date (October 2008), and links to errata and previous versions. A version selection table shows RFC 5321 as the current version. The bottom of the page indicates it is part of the Standards Track and is page 61 of the document.

PROTOCOL = "ESMTP" / "SMTP" / ATTDL-PROTOCOL

Attdl-Protocol = Atom

; Additional standard names for protocols are
 ; registered with the Internet Assigned Numbers
 ; Authority (IANA) in the "mail parameters"
 ; registry [9]. SMTP servers SHOULD NOT
 ; use unregistered names.

4.5. Additional Implementation Issues

4.5.1. Minimum Implementation

In order to make SMTP workable, the following minimum implementation MUST be provided by all receivers. The following commands MUST be supported to conform to this specification:

EHLO
 HELO
 MAIL
 RCPT
 DATA
 RSET
 NOOP
 QUIT
 VRFY

Any system that includes an SMTP server supporting mail relaying or delivery MUST support the reserved mailbox "postmaster" as a case-insensitive local name. This postmaster address is not strictly necessary if the server always returns 554 on connection opening (as described in [Section 3.1](#)). The requirement to accept mail for postmaster implies that RCPT commands that specify a mailbox for postmaster at any of the domains for which the SMTP server provides mail service, as well as the special case of "RCPT TO:<Postmaster>" (with no domain specification), MUST be supported.

SMTP systems are expected to make every reasonable effort to accept mail directed to Postmaster from any other system on the Internet. In extreme cases -- such as to contain a denial of service attack or other breach of security -- an SMTP server may block mail directed to Postmaster. However, such arrangements SHOULD be narrowly tailored so as to avoid blocking messages that are not part of such attacks.

Document type
 RFC **Draft Standard**
 October 2008
[View errata](#)
[Report errata](#)
 Updated by [RFC 7504](#)
 Obsoletes [RFC 2821](#)
 Updates [RFC 1123](#)
 Was [draft-klensin-rfc2821bis](#) (individual in app area)

Select version

00	01	02	03	04
05	06	07	08	09
10	11	RFC 5321		

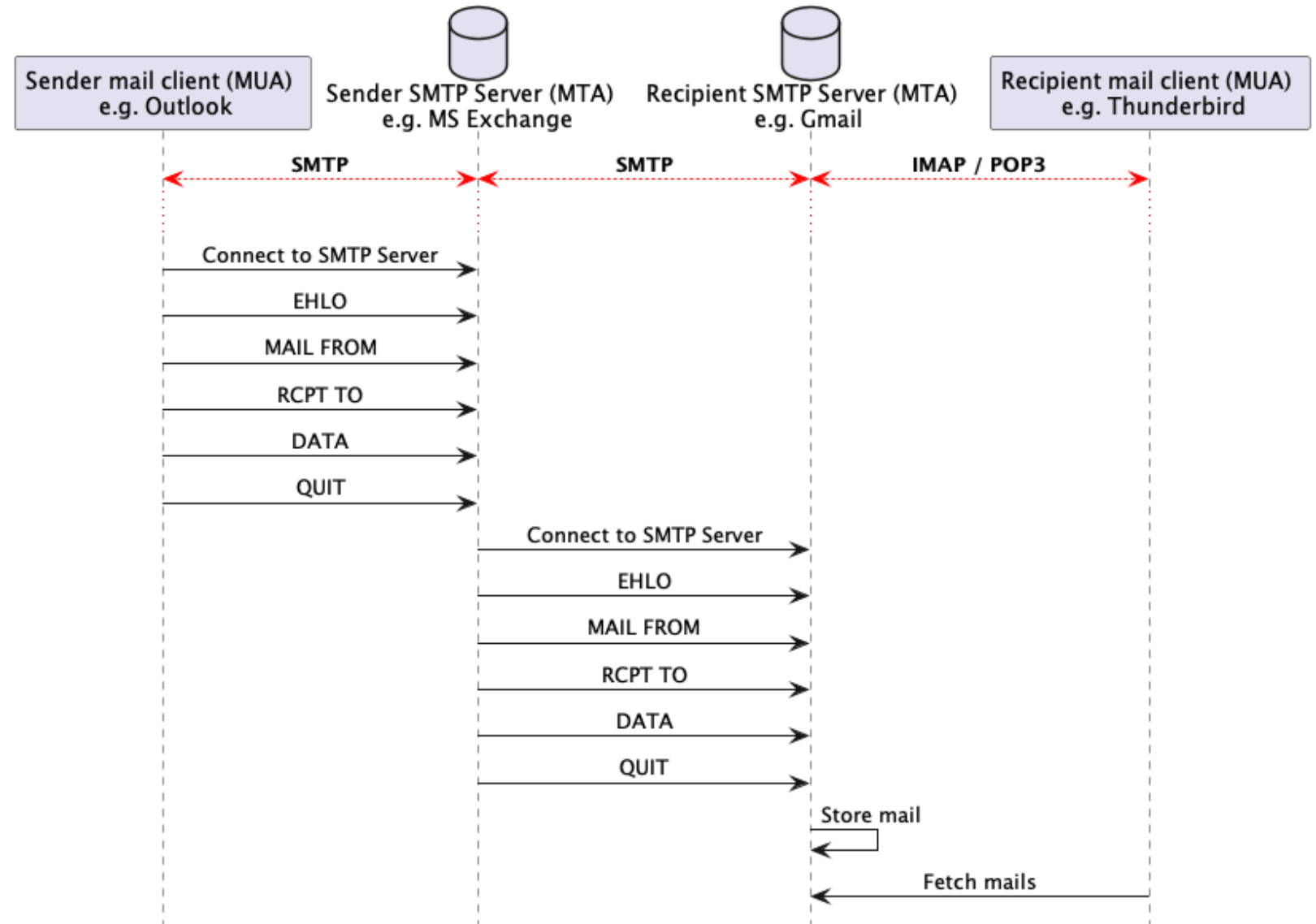
Compare versions
 ...nsin-rfc2821bis-10
 RFC 5321
[Side-by-side](#) [Inline](#)

Author
[Dr. John C. Klensin](#) ✉
[Email author](#)

Klensin Standards Track [Page 61]

SMTP and ncat

- HELO / EHLO
- MAIL FROM
- RCPT TO
- DATA
 - Subject:
 - From:
 - To:
 - End by .
- QUIT

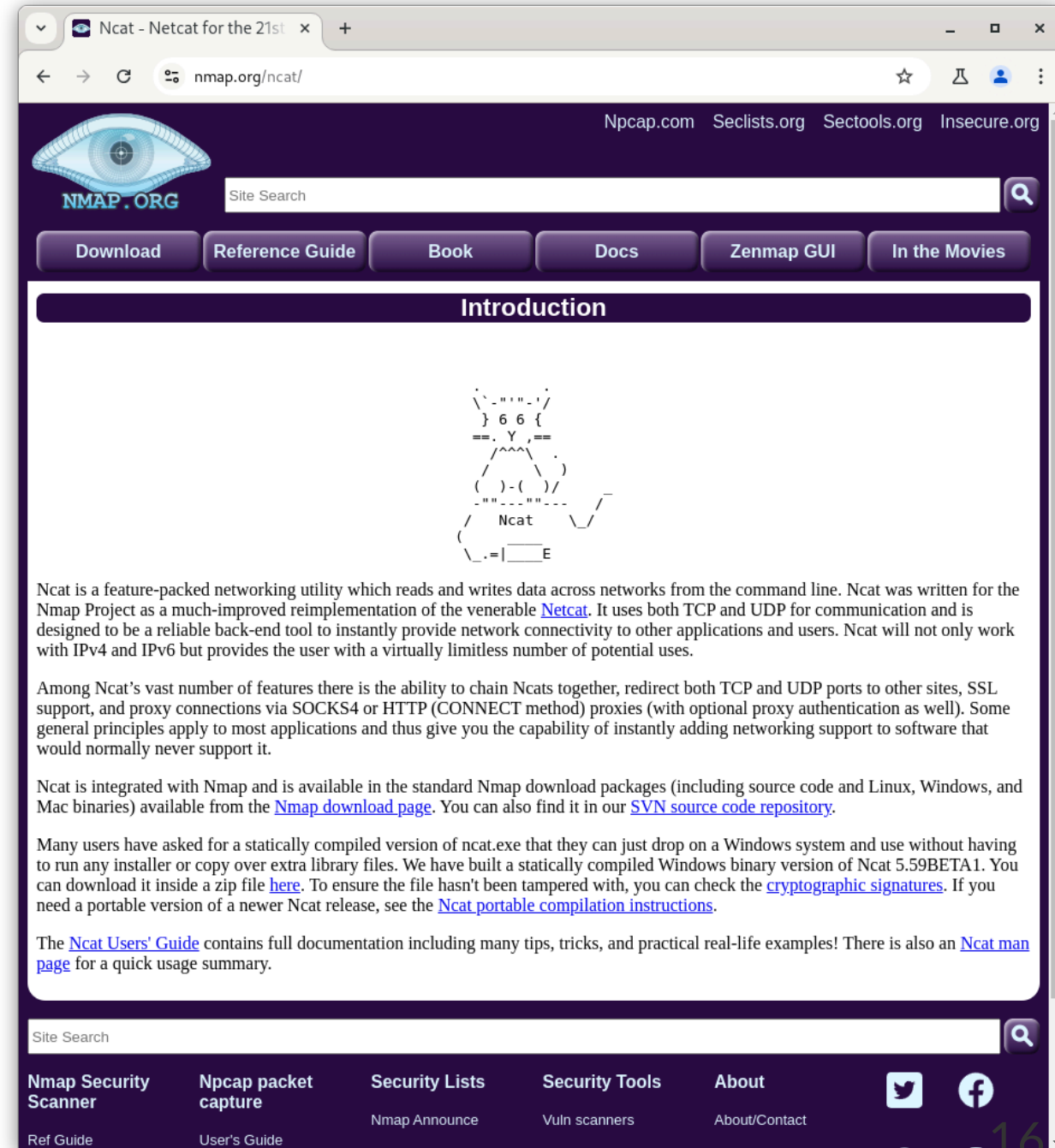


ncat

More details for this section in the [course material](#). You can find other resources and alternatives as well.

ncat

- ncat is network utility for reading from and writing to network connections
- It is used to connect to a remote server (SMTP, HTTP, ...)
- We will use it to interact with a SMTP server



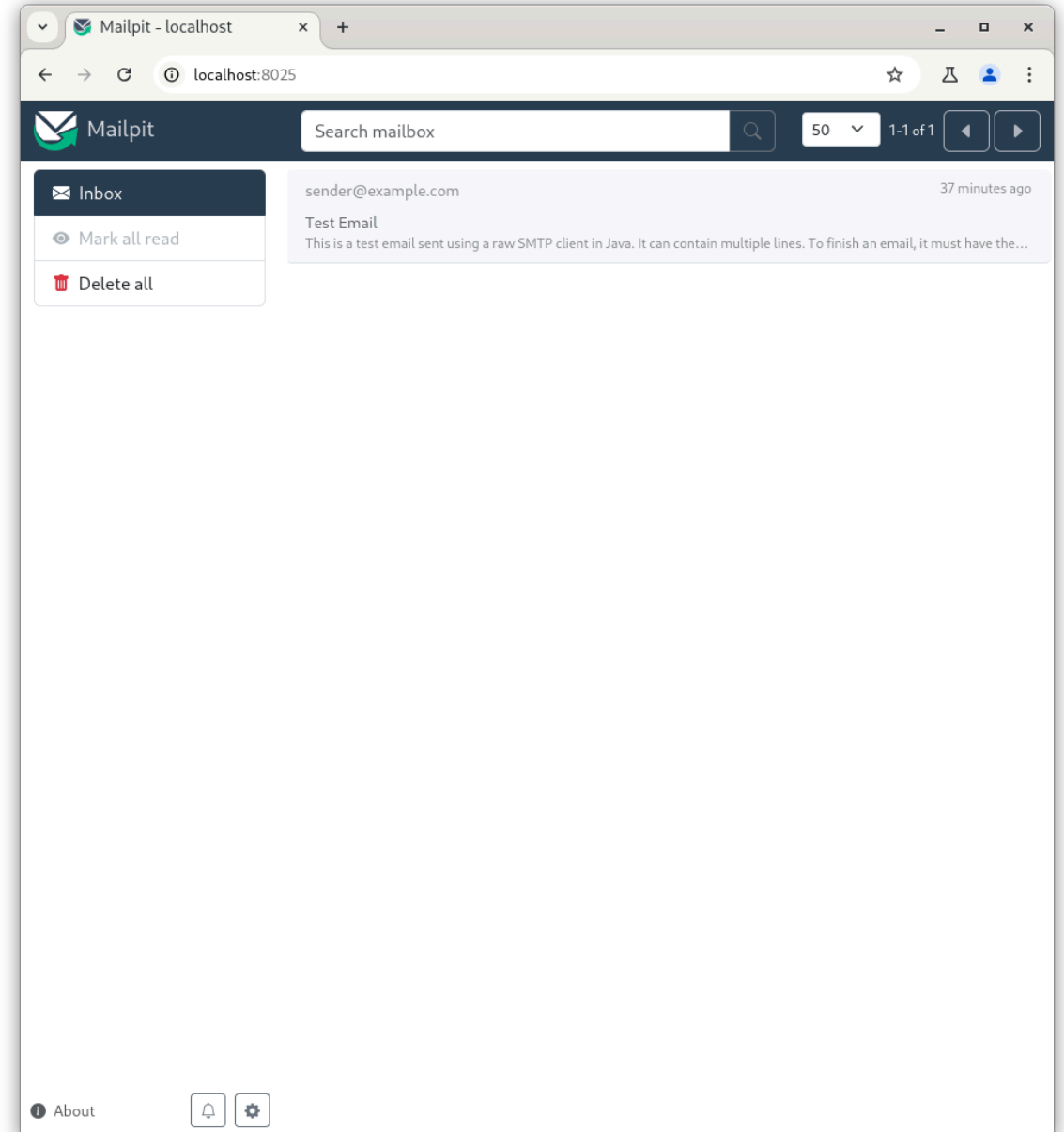
Questions

Do you have any questions?

Practical content

What will you do?

- Install and configure ncat
- Start a SMTP server with Docker Compose
- Send an email with ncat to the SMTP server
- Send an email with Java to the SMTP server



Now it's your turn!

- Read the course material.
- Do the practical content.
- Ask questions if you have any.

 [Find the course on GitHub.](#)

Do not hesitate to help each other! There's no need to rush!



Finished? Was it easy? Was it hard?

Can you let us know what was easy and what was difficult for you during this chapter?

This will help us to improve the course and adapt the content to your needs. If we notice some difficulties, we will come back to you to help you.

 [GitHub Discussions](#)

You can use reactions to express your opinion on a comment!

What will you do next?

We are arriving at the end of the second part of the course.

An evaluation will be done to check your understanding of all the content seen in this second part.

More details will be given in the next chapter.



Sources

- Main illustration by [Joanna Kosinska](#) on [Unsplash](#)
- Illustration by [Aline de Nadai](#) on [Unsplash](#)
- Illustration by [Nik](#) on [Unsplash](#)
- Illustration by [MChe Lee](#) on [Unsplash](#)