Networking Basics

07a - Simple Mail Transfer Protocol (SMTP)

Wolfgang Tremmel
academy@de-cix.net

Where networks meet

www.de-cix.net

Networking Basics

DE-CIX Academy

01 - Networks, Packets, and Protocols

02 - Ethernet, 02a - VLANs

03 - IP, 03a - Routing, 03b - Global routing

04a - UDP   04b - TCP   04c - ICMP

05 - Uni-, Broad-, Multi-, and Anycast

06a - Domain Name System (DNS)

→ 07a - Simple Mail Transfer Protocol (SMTP)

Internet Model

IP / Internet Layer

- Data units are called "Packets"
- Provides source to destination transport
  - For this we need addresses
- Examples:
  - IPv4
  - IPv6

Layer	Name
5	Application
4	Transport
3	Internet
2	Link
1	Physical

Internet Model

Transport Layer

- *May* provide flow control, reliability, congestion avoidance
- Also may contain information about the next layer up
- Examples:
    - UDP (none of the above)
    - TCP (flow control, reliability, congestion avoidance)

Layer	Name
5	Application
4	Transport
3	Internet
2	Link
1	Physical

Internet Model

Application Layer

Layer	Name
5	Application
4	Transport
3	Internet
2	Link
1	Physical

- Depends on transport layer
  - by using either UDP or TCP as transport
- Contains communications protocols and interfaces used in process-to-process communication across IP networks
- Both client-server and peer-to-peer relationships are possible
- Many examples: Email, Web, Audio, Video...

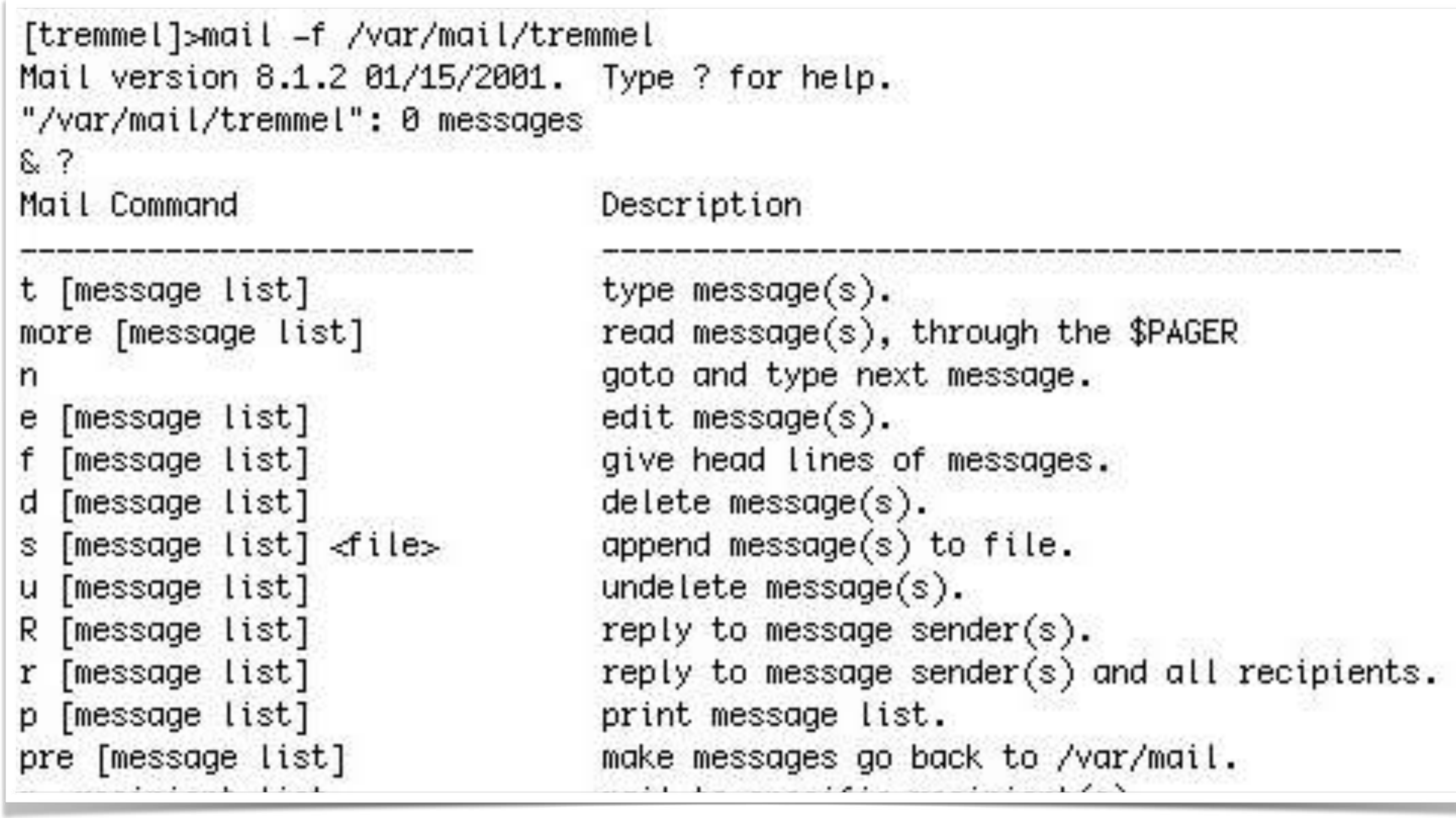
Application Layer: EMail

DE CIX

EMail

Application Layer

- One of the oldest (still in use) applications of Internet
- RFCs mentioning "mail": 483
- So this presentation does not cover everything
- it focuses on email **transfer**
- There is a protocol for that...



```
[tremmel]>mail -f /var/mail/tremmel
Mail version 8.1.2 01/15/2001.  Type ? for help.
"/var/mail/tremmel": 0 messages
& ?
Mail Command                    Description
-------------------------       --------------------------------------------
t [message list]                type message(s).
more [message list]             read message(s), through the $PAGER
n                               goto and type next message.
e [message list]                edit message(s).
f [message list]                give head lines of messages.
d [message list]                delete message(s).
s [message list] <file>         append message(s) to file.
u [message list]                undelete message(s).
R [message list]                reply to message sender(s).
r [message list]                reply to message sender(s) and all recipients.
p [message list]                print message list.
pre [message list]              make messages go back to /var/mail.
```

SMTP

SMTP

SMTP

Simple

SMTP

Simple Mail

SMTP

Simple Mail Transfer

SMTP

Simple Mail Transfer Protocol

SMTP

Simple Mail Transfer Protocol

- Introduced in 1981
- RFC788
- Was there also a non-simple protocol?
  - Yes - RFC772 - Mail Transfer Protocol
- Latest standard: RFC5321 (2008) 94 pages long

Attribution: National Postal Museum, Philatelic Collection. Library and Archives Canada, S-003512 / Musée national de la poste, collection philatélique. Bibliothèque et Archives Canada, S-003512

SMTP

Simple Mail Transfer Protocol

- What does it do?
  - Transfer email
  - From servers to servers
  - From users to servers
    - This is also called "submission" and may use a different port

SMTP

Port numbers

- **TCP port 25**
  - Standard port from the beginning
- For submission: TCP port 587
  - Submission (user to server) might have different requirements
- 465 - for encrypted submission

Attribution: https://digitaltmuseum.se/011014704473/tomning-av-brevlada-pa-timmermansgatan-invid-sodra-station

Email

Email

Structure

Poſtkarte
Verlag Friedr. Wunder, Schifferstadt. Ges. geschützt. Nachdruck verboten
SCHIFFERSTADT BAYERN
7-8 Nm
11

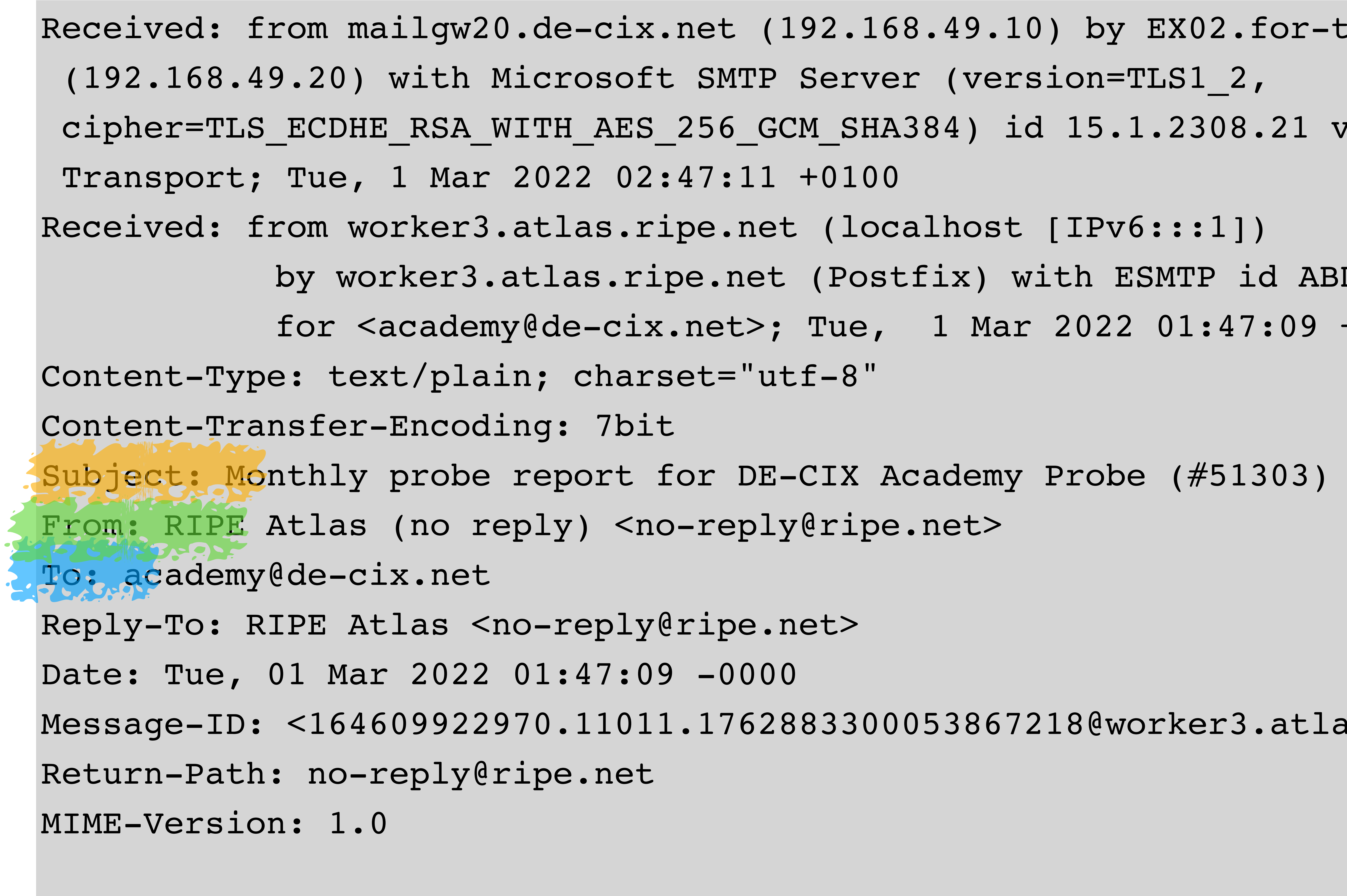
Postcard from 1911 - in possession of the author

- "Visible" components:
    - Header
        - Contains lines formatted like "Field name: Field content"
        - Like: "Subject:", "From:", "To:", "Message-ID:"
        - Most of it is hidden by your email client, but can be made made visible
    - Body
- "Invisible" component
    - Envelope

Email

Header

- Contains lines formatted like *Field name: Field content*
  - Like: "Subject:", "From:", "To:", "Message-ID:"
  - Most of it is hidden by your email client, but can be made made visible
  - Check your email client documentation



```
Received: from mailgw20.de-cix.net (192.168.49.10) by EX02.for-t
 (192.168.49.20) with Microsoft SMTP Server (version=TLS1_2,
 cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id 15.1.2308.21 v
 Transport; Tue, 1 Mar 2022 02:47:11 +0100
Received: from worker3.atlas.ripe.net (localhost [IPv6:::1])
          by worker3.atlas.ripe.net (Postfix) with ESMTP id ABI
          for <academy@de-cix.net>; Tue,  1 Mar 2022 01:47:09 -
Content-Type: text/plain; charset="utf-8"
Content-Transfer-Encoding: 7bit
Subject: Monthly probe report for DE-CIX Academy Probe (#51303)
From: RIPE Atlas (no reply) <no-reply@ripe.net>
To: academy@de-cix.net
Reply-To: RIPE Atlas <no-reply@ripe.net>
Date: Tue, 01 Mar 2022 01:47:09 -0000
Message-ID: <164609922970.11011.1762883300053867218@worker3.atla
Return-Path: no-reply@ripe.net
MIME-Version: 1.0
```

Email

Body

- No binary! Only "textual" content
- But I can email photos!
    - Yes, they are encoded using the MIME standard
- Usually base64 is used for encoding

```
MIME-Version: 1.0

--0000000000004a26c305d88616ef
Content-Type: text/plain; charset="UTF-8"; format=flowed; delsp=
Content-Transfer-Encoding: base64

SmFuIFpvcnogcmVwbGllZCB0byBhIGNvbW1lbnQgaW4gdGhlIGZvbGxvd2luZyBk
UklQRSA4NCBDZlAgIA0KKGh0dHBzOi8vZG9jcy5nb29nbGUuY29tL2RvY3VtZW50
```

Attribution: Archives Reference: AAME 8106 W5603 Box 126 archway.archives.govt.nz/ViewFullItem.do?code=20940969

Email

Envelope

- Data mail servers exchange before transmitting an email
- Minimum:
  - Name of sending server
  - Email address of originator
  - Email address of recipient (one or more)
- Optional:
  - Size of email
  - other stuff

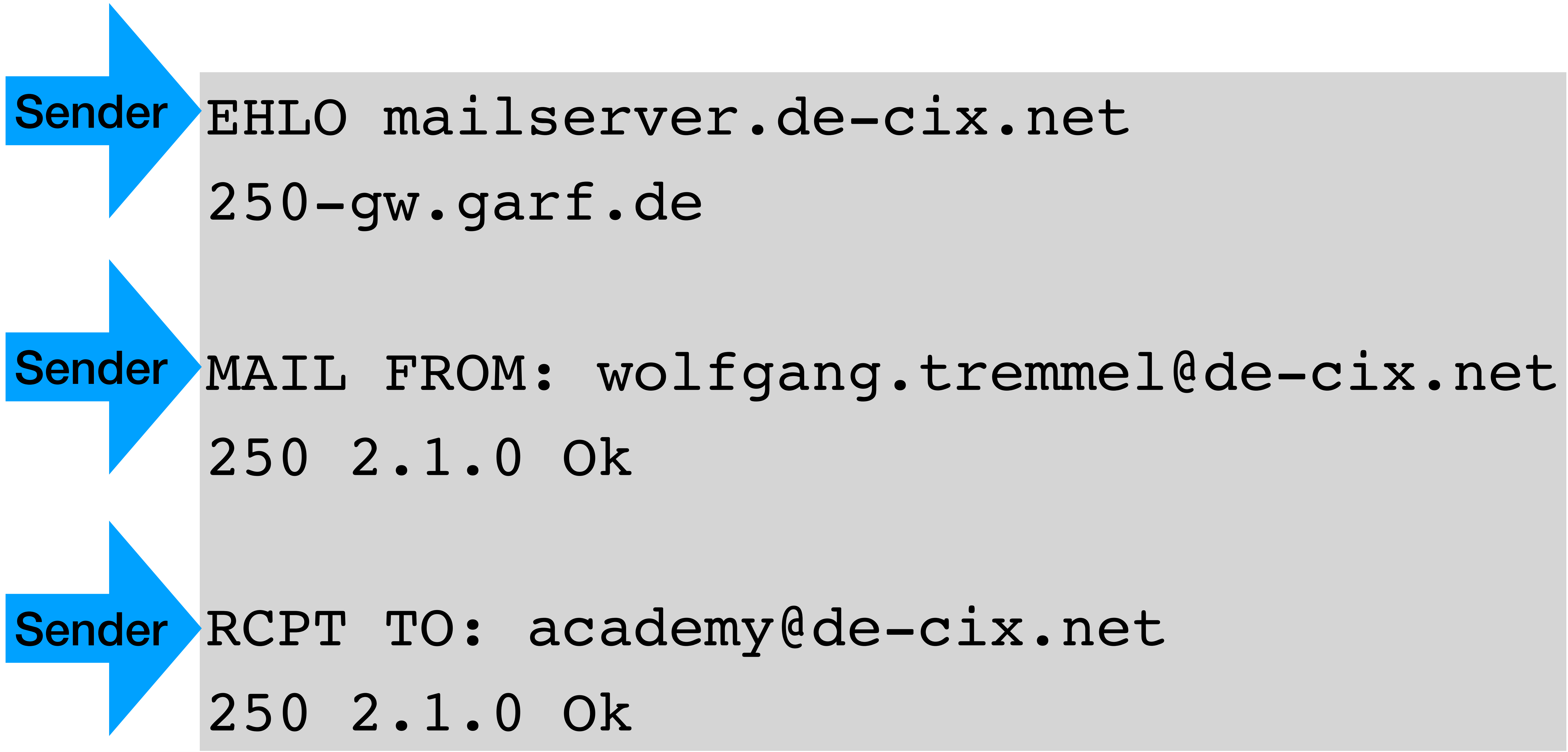
```
EHLO mailserver.de-cix.net
250-gw.garf.de

MAIL FROM: wolfgang.tremmel@de-cix.net
250 2.1.0 Ok

RCPT TO: academy@de-cix.net
250 2.1.0 Ok
```

Attribution: Mail receiving and sorting room, U.S. Post Office, Washington, D.C.) - National Photo Co., Washington, D.C LCCN90709850.jpg

SMTP

The protocol

- Lets keep this example
- All SMTP interactions can be read as text
- The sender uses commands (used to be 4 letters long)
  - Why start with "EHLO"?
  - Originally it was "HELO"
- The receiver answers with 3-digit error/success codes



```
EHLO mailserver.de-cix.net
250-gw.garf.de

MAIL FROM: wolfgang.tremmel@de-cix.net
250 2.1.0 Ok

RCPT TO: academy@de-cix.net
250 2.1.0 Ok
```

- You might have guessed
  - "250" means "ok, no error"

SMTP

Commands

- First the sender identifies it self using "EHLO *hostname*"
  - The receiver answers with a 3-digit status code and a list of capabilities
  - Do you notice the "-" between the status code and the capability?
  - It is missing at the last line, marking it the end of a multi line reply.

```
EHLO sender.example.com
250-receiver.example.net
250-SIZE 204800000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-DSN
250-SMTPUTF8
250 CHUNKING
```

SMTP

Commands

- Then the sender starts an email transmission

- MAIL FROM is the envelope address of the sender of the email.

- RCPT TO gives one recipient of the email. Can be repeated if multiple recipients.

- DATA starts the transmission of the email

- a "." alone on a line marks the end of the email transmission (there is a procedure so the content of any email can contain a dot alone on a line).

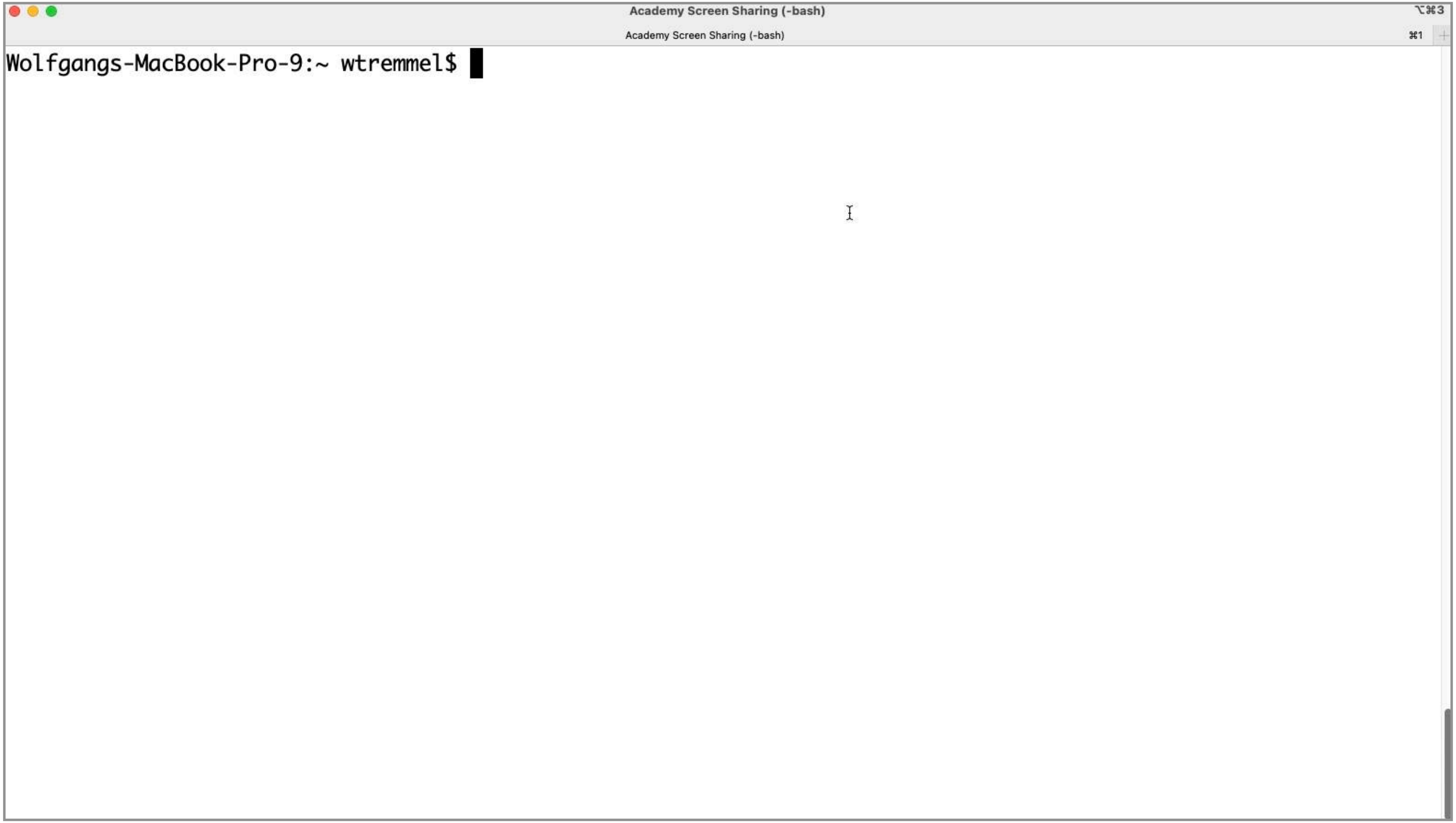
```
MAIL FROM: academy@de-cix.net
250 2.1.0 OK
RCPT TO: someone@example.com
250 2.1.5 OK
DATA
354 End data with <CR><LF>.<CR><LF>
...some email...
.
250 2.0.0 Ok: queued as 000121
QUIT
221 2.0.0 Bye
```

SMTP

Commands

- After transmitting an email the sender can either transmit the next one...
  - starting again with "MAIL FROM"
  - or QUIT
- The receiver sends 3-digit status codes:
  - 250:
    - 2 stands for "positive completion"
    - 5 stands for "from the mail system"
    - ("0" is "no additional information")

```
MAIL FROM: academy@de-cix.net
250 2.1.0 OK
RCPT TO: someone@example.com
250 2.1.5 OK
DATA
354 End data with <CR><LF>.<CR><LF>
...some email...
.
250 2.0.0 Ok: queued as 000121
QUIT
221 2.0.0 Bye
```


Academy Screen Sharing (-bash)
⌥⌘3
Academy Screen Sharing (-bash)
⌘1

```
Wolfgangs-MacBook-Pro-9:~ wtremmel$ 
```

This does not look very secure, right?

SMTP

(missing any) Security

- When the Internet was young, nobody cared much about security
- Everybody trusted each other
- SMTP is from that time
- Email senders can (still) be easily faked
- Any system could (and can) send any email to any recipient.

U.S. MAIL
RURAL ROUTE
HORSE DRAWN WAGON CIRCA 1898 USED TO DELIVER MAIL TO RURAL AREAS

Attribution: Jim Henderson, https://commons.wikimedia.org/wiki/File:US_Mail_Coach_jeh.jpg

Adding security to SMTP

Features added

- Encrypted transmission
  - Command STARTTLS was added 1999 in RFC2487
  - It protects just the transmission of the email, **not** the content
- Authentication
  - AUTH command was added in 1999 in RFC2554
  - Mainly used for username/password authentication to send email
  - So mail server users can authenticate themselves

Attribution: A secure rural mail box by Stanley Howe
https://commons.wikimedia.org/wiki/File:A_secure_rural_mail_box_-_geograph.org.uk_-_2557317.jpg

Conclusion

Conclusion

SMTP + Email

- Simple Mail Transfer Protocol is a text-based protocol on the application layer
  - SMTP is "spoken" via TCP on port 25
- It is one of the oldest still in use protocols
- Over the years it has been extended multiple times
- It is highly recommended to enable all security features like TLS and authentication
- Emails themselves can be faked easily, unless you use additional features like cryptographic signatures.

Thank you!

academy@de-cix.net

Interested in more webinars? Please subscribe to our mailing list at https://lists.de-cix.net/wws/subscribe/academy

Links and further reading

Links and further reading

- Internet protocol - https://en.wikipedia.org/wiki/Internet_Protocol
- Protocol stack - https://en.wikipedia.org/wiki/Protocol_stack
  - Transport Layer: https://en.wikipedia.org/wiki/Transport_layer
  - Datagram: https://en.wikipedia.org/wiki/Datagram
- IP Network Model: https://en.wikipedia.org/wiki/Internet_protocol_suite
- IPv4
  - IPv4 - https://en.wikipedia.org/wiki/IPv4
- IPv6
  - IPv6 itself - https://en.wikipedia.org/wiki/IPv6
  - IPv6 header - https://en.wikipedia.org/wiki/IPv6_packet
- History of Internet and IP
  - Internet Hall of Fame - https://internethalloffame.org
  - Defense Advanced Research Projects Agency (DARPA) - https://www.darpa.mil
  - ARPANET - https://www.darpa.mil/about-us/timeline/arpanet
  - The "Protocol Wars" - https://en.wikipedia.org/wiki/Protocol_Wars

Links and further reading

- Mail transfer:
  - Mail transfer protocol: RFC772 (ancient history)
  - Simple mail transfer protocol:
    - First RFC on SMTP: RFC788
    - A long time in use was RFC821 (valid from 1982 until 2001)
    - Most recent standard: RFC5321 (October 2008)
  - All RFCs related to SMTP would be too many to list here, simply search for them.

- Message submission:
  - Introduced 1998 in RFC2476
  - Current standard: RFC6409 (with some updates, check yourself)

- Transport Layer Security (TLS) for email:
  - Introduced 1999 in RFC2487
  - Current standard: RFC3207 (there are updates - check yourself)

Links and further reading

- Mail encoding:
  - MIME standard: https://en.wikipedia.org/wiki/MIME
  - MIME in emails: RFC2045 , RFC2046, RFC2047 (there are more...)
  - BASE64 encoding: RFC4648
  -