Agenda

10:10 📝 Internet Message Format & Quick introduction to email delivery

🔨 Challenges - Volume 1

11:00 ☕ Break

11:10 📝 Email Authentication & Triaging email abuse

🔨 Challenges - Volume 2

slides: bzcat.eu/email/

challenges: ctf.bzcat.eu

Internet Message Format

- ~~RFC 822 - August 1982~~
- ~~RFC 2822 - April 2001~~
- **RFC 5322 - October 2008**
  - https://datatracker.ietf.org/doc/html/rfc5322

RFC 2822

```
From: John Doe
<jdoe@machine.example>
To: Mary Smith <mary@example.net>
Subject: Saying Hello
Date: Fri, 21 Nov 1997 09:55:06 -
0600
Message-ID:
<1234@local.machine.example>

This is a message just to say
hello.
So, "Hello".
```

Header, Header, Header and Header

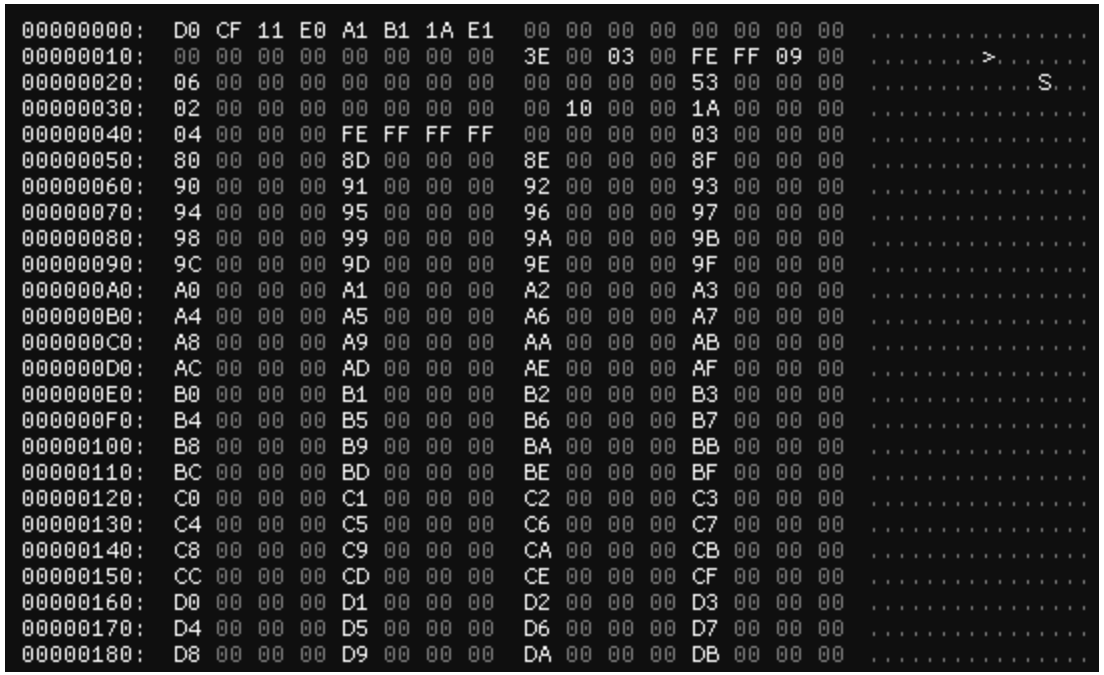
Field	Min number	Max number	Notes
trace	0	unlimited	Block prepended - see
			3.6.7
resent-date	0*	unlimited*	One per block, required if
			other resent fields are
			present - see 3.6.6
resent-from	0	unlimited*	One per block - see 3.6.6
resent-sender	0*	unlimited*	One per block, MUST occur
			with multi-address
			resent-from - see 3.6.6

RFC2045, 2046, 2047, 2049: Multipurpose Internet Mail Extensions

- RFC 2045: Format of Internet Message Bodies
- RFC 2046: Media Types
- RFC 2047: Message Header Extensions for Non-ASCII Text
- RFC 2048: Registration Procedures
- RFC 2049: Conformance Criteria and Examples

```
From: Moderator-Address
To: Recipient-List
Date: Mon, 22 Mar 1994 13:34:51 +0000
Subject: Internet Digest, volume 42
MIME-Version: 1.0
Content-Type: multipart/mixed;
              boundary="---- main boundary ----"

------ main boundary ----

  ...Introductory text or table of contents...

------ main boundary ----
 Content-Type: multipart/digest;
              boundary="---- next message ----"

------ next message ----

From: someone-else
 Date: Fri, 26 Mar 1993 11:13:32 +0200
 Subject: my opinion

  ...body goes here ...

------ next message ----

From: someone-else-again
 Date: Fri, 26 Mar 1993 10:07:13 -0500
 Subject: my different opinion

  ... another body goes here ...

------ next message ------

------ main boundary ------
```

CDFV2 Microsoft Outlook Message



CDFV2 Microsoft Outlook Message - Reading

- Convert to EML

```
apt install libemail-outlook-message-perl
msgconvert message.msg
```

- EML_analyzer does it automatically

EML Analyzer

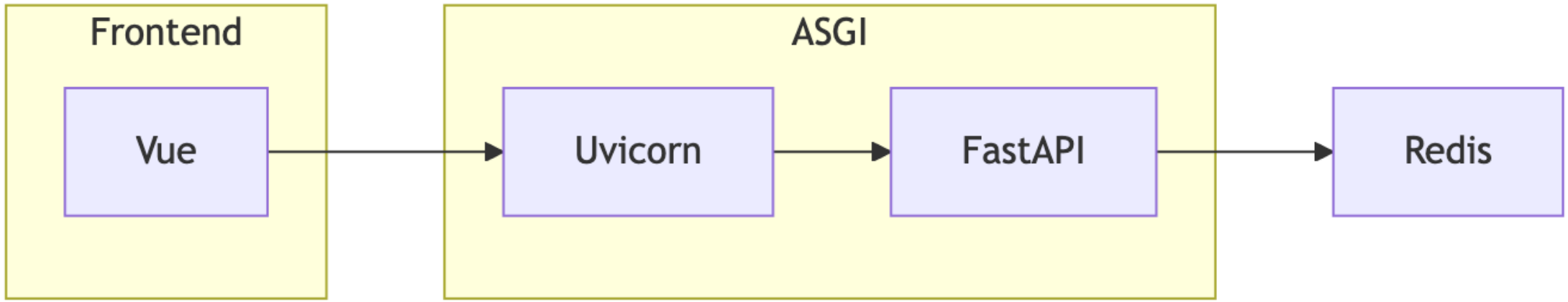
- Parse an EML/MSG file as a JSON

```
From: foo@example.com
To: bar@example.com
Subject: test
Date: Sun, 8 Jan 2017 20:37:44 +0200

Hello world!
```

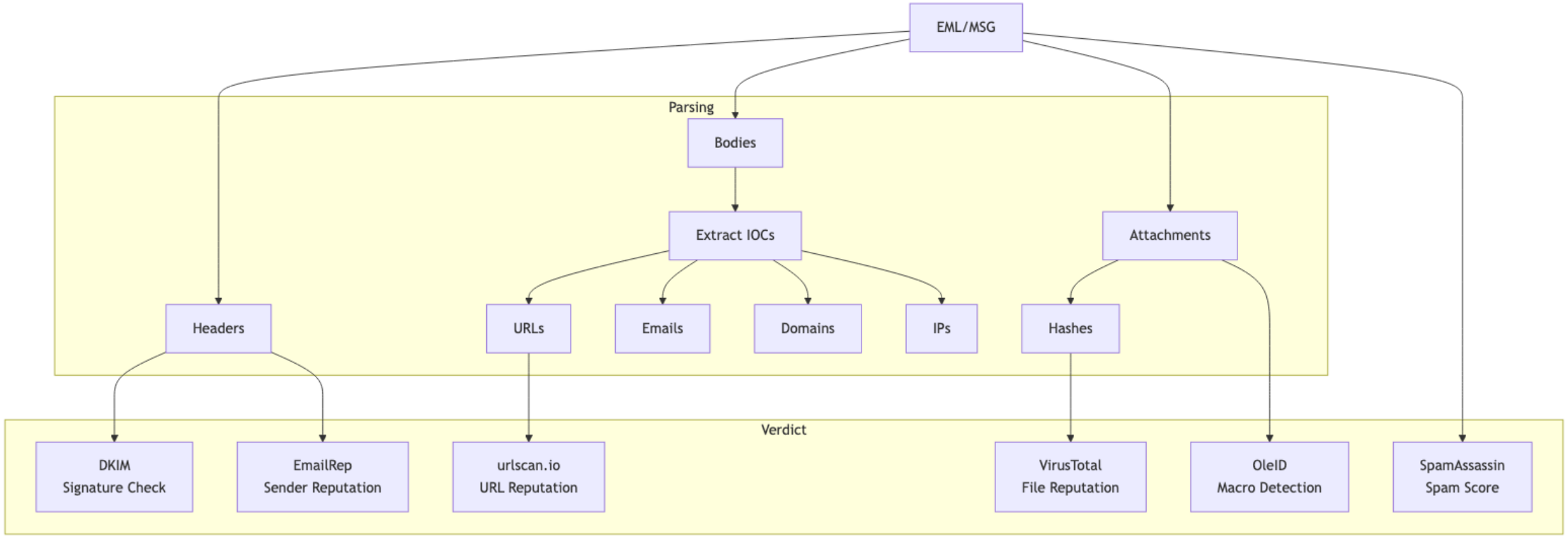
```
{
    "eml": {
        "attachments": [],
        "bodies": [
            {
                "contentType": null,
                "hash": "0ba904eae8773b70c75333db4de2f3ac45a8ad4ddba1b242f0b3cfc199391dd8",
                "contentHeader": {},
                "content": "Hello world!\n",
                "urls": [],
                "emails": [],
                "domains": [],
                "ipAddresses": []
            }
        ],
        "header": {
            "messageId": null,
            "subject": "test",
            "defect": null,
            "from": "foo@example.com",
            "to": [
                "bar@example.com"
            ],
            "cc": null,
            "date": "2017-01-08T20:37:44+02:00",
            "receivedEmail": null,
            "receivedForemail": null,
            "receivedDomain": null,
            "receivedIp": null,
            "receivedSrc": null,
            "received": [],
            "header": {
                "date": [
                    "2017-01-08T20:37:44+02:00"
                ],
                "from": [
                    "foo@example.com"
                ],
                "subject": [
                    "test"
                ],
                "to": [
                    "bar@example.com"
                ]
            }
        }
    },
    "verdicts": [
        {
            "name": "SpamAssassin",
            "malicious": false,
            "score": 0.1,
            "details": [
                {
                    "key": "MISSING_MID",
                    "score": 0.1,
                    "description": "Missing Message-Id: header",
                    "referenceLink": null
                },
                {
                    "key": "NO_RECEIVED",
                    "score": -0.0,
                    "description": "Informational: message has no Received headers",
                    "referenceLink": null
                },
                {
                    "key": "NO_RELAYS",
                    "score": -0.0,
                    "description": "Informational: message was not relayed via SMTP",
                    "referenceLink": null
                }
            ]
        },
        {
            "name": "oleid",
            "malicious": false,
            "score": null,
            "details": [
                {
                    "key": "benign",
                    "score": null,
                    "description": "There is no suspicious OLE file in attachments.",
                    "referenceLink": null
                }
            ]
        }
    ],
    "id": "9561332a339eea1209da050f1cb86ecc75404082a1a37d2c65e029f28ad694c9"
}
```

- `eml`: parsed EML message
- `verdicts`: verdicts by 3rd party services (Spam Assassin, oletools, urlscan.io, VT, etc.)
- `id`: SHA256 hash of an EML message

System Overview



Data Flow


EML/MSG
Parsing
Bodies
Extract IOCs
Attachments
Headers
URLs
Emails
Domains
IPs
Hashes
Verdict
DKIM
Signature Check
EmailRep
Sender Reputation
urlscan.io
URL Reputation
VirusTotal
File Reputation
OleID
Macro Detection
SpamAssassin
Spam Score

Standing on the shoulders of giants

- GOVCERT-LU/eml_parser/: python eml parser module
- JoshData/convert-outlook-msg-file: Python library to convert Microsoft Outlook .msg files to .eml/MIME message files.

Use Cases

- You Move, They Follow Uncovering Iran’s Mobile Legal Intercept System
- FIESTA 2023 Writeup

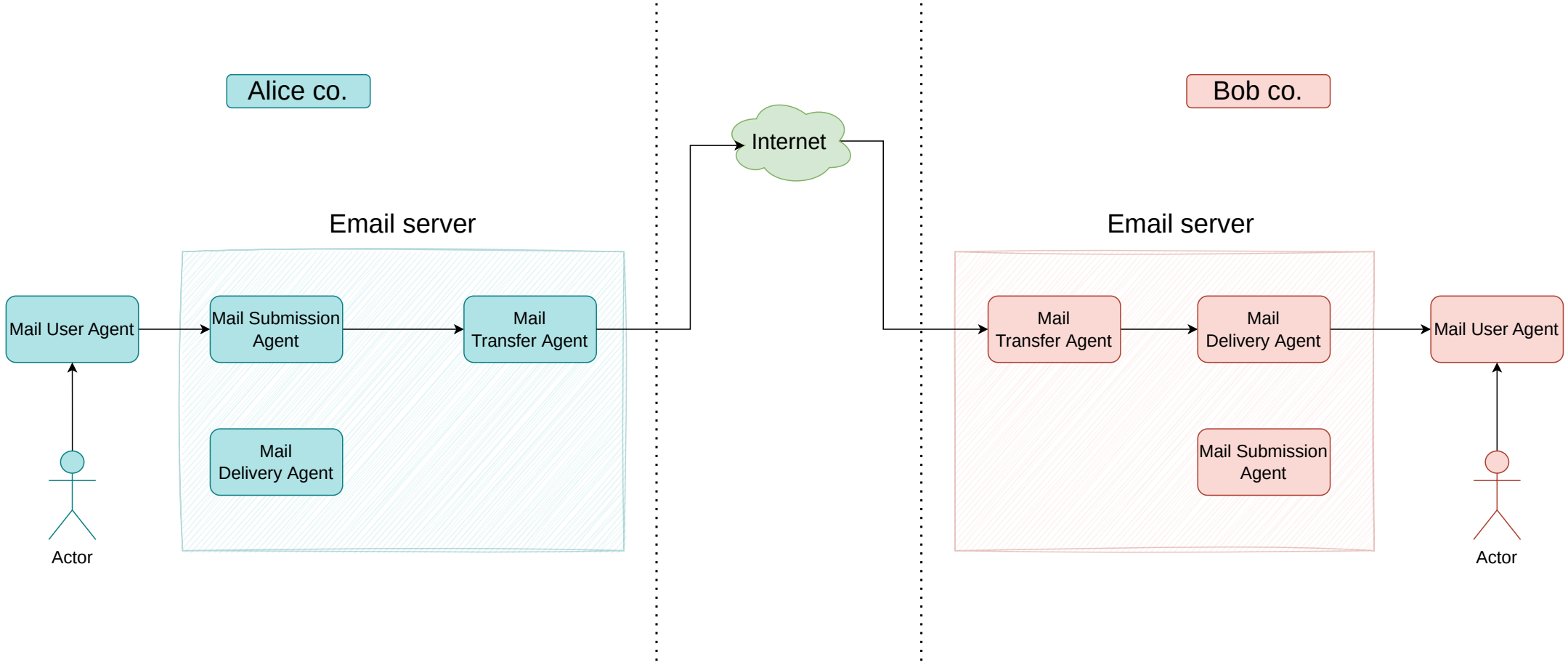
EML Analyzer Up & Running

```
git clone https://github.com/ninoseki/eml_analyzer/
cd eml_analyzer
docker compose up -d
```

Open `localhost:8000`.

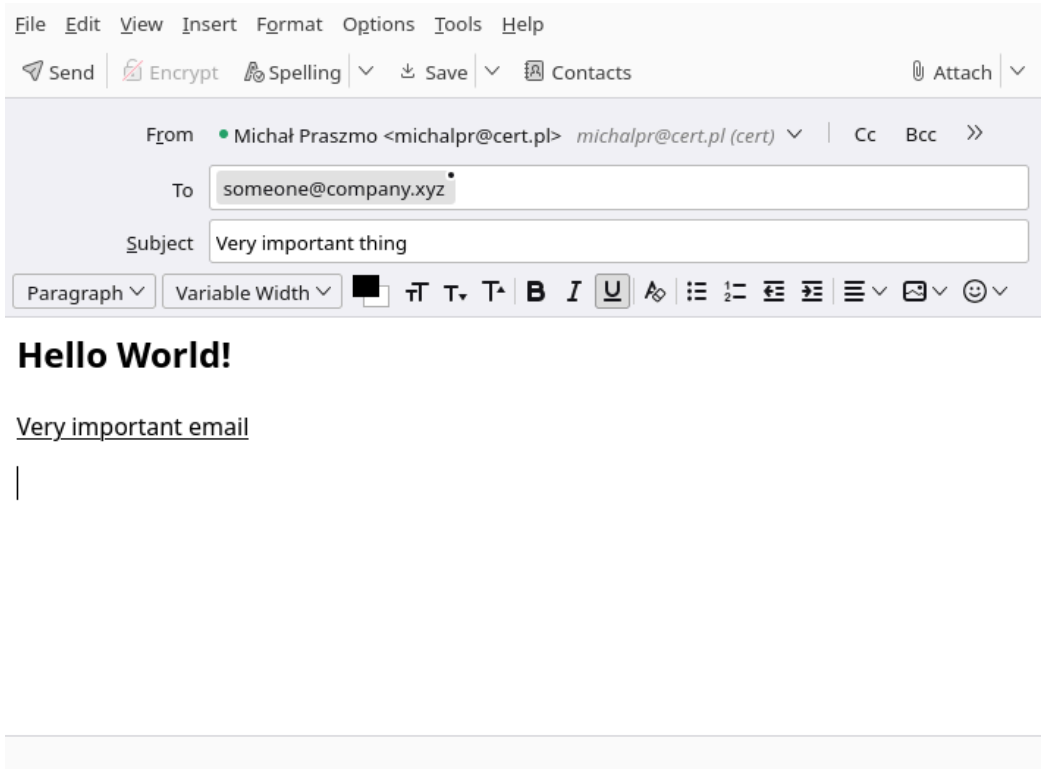
Quick introduction to email delivery

Life cycle of email delivery



Mail User Agent - MUA

User-friendly client to compose emails and submit them to the email server



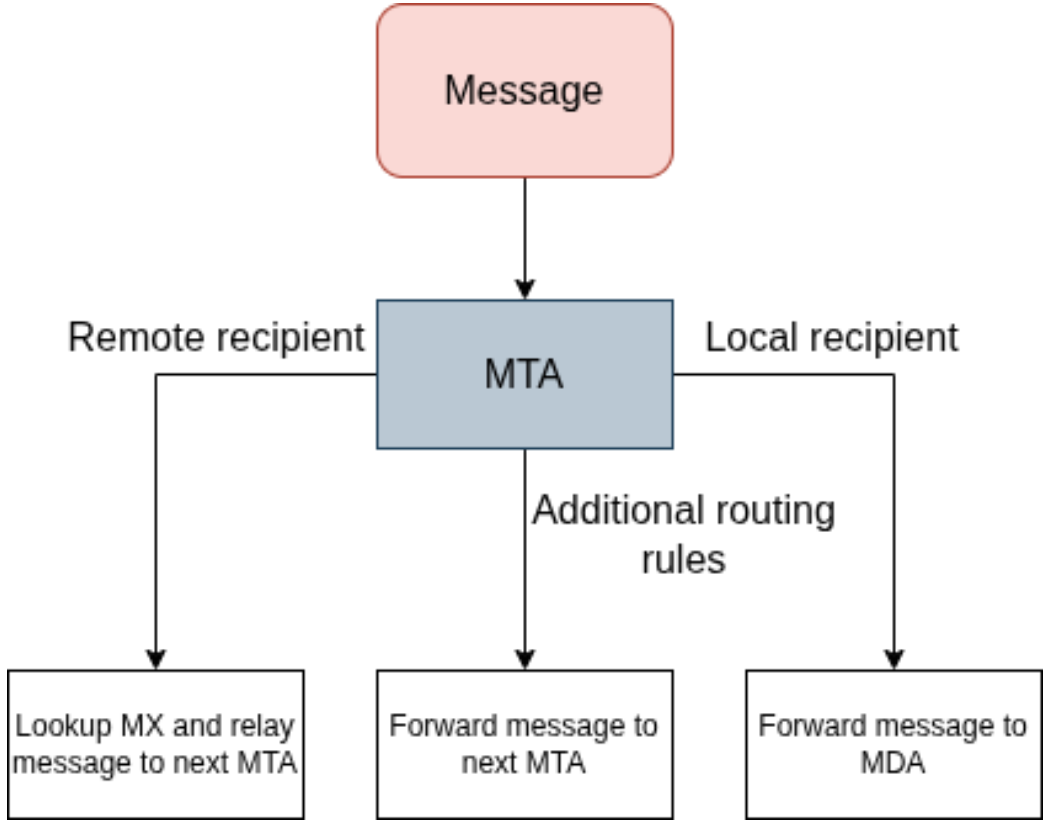
Mail Submission Agent - MSA

- Required authentication
- Message validation & correction
- Sometimes a part of MTA

Mail Transfer Agent - MTA

Outgoing messages

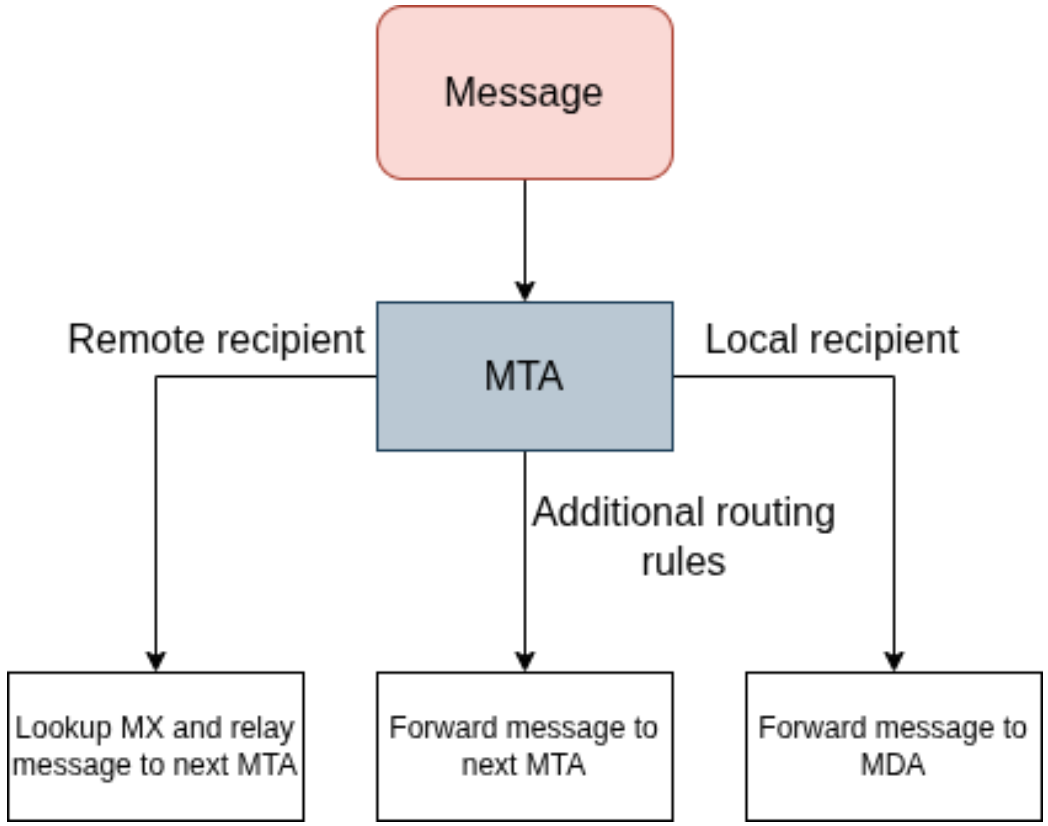
- Queueing
- Routing
- Handling bounces
- Adding security headers



Mail Transfer Agent - MTA

Incoming messages

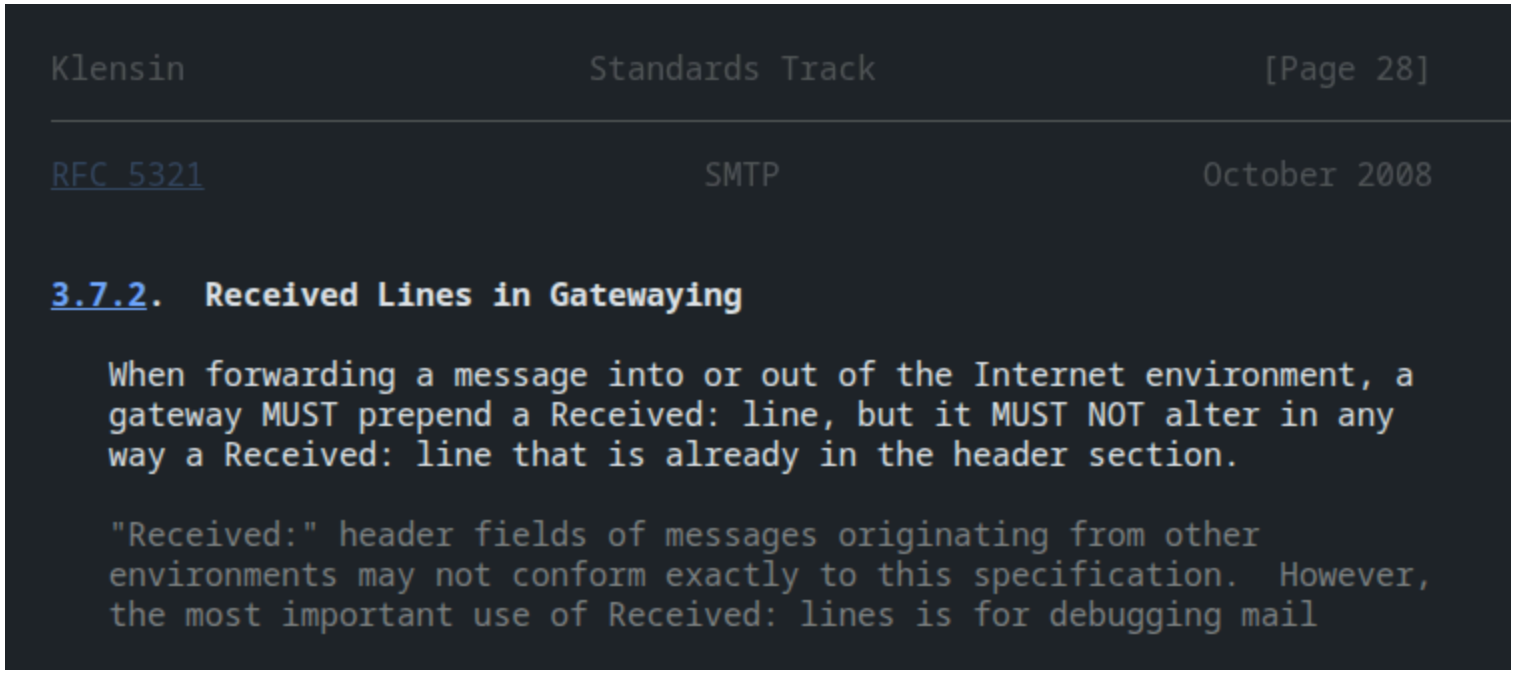
- Verifying security headers (SPF/DKIM/DMARC/ARC)
- Checking if the recipient is internal or external (relay)
- Monitoring sender reputation (spam filtering)



Mail Delivery Agent - MDA

- Save email message to user's mailbox
- Additional mail filtering
- User-specified filtering rules

How to make sense out of Mail relays



Klensin Standards Track [Page 28]

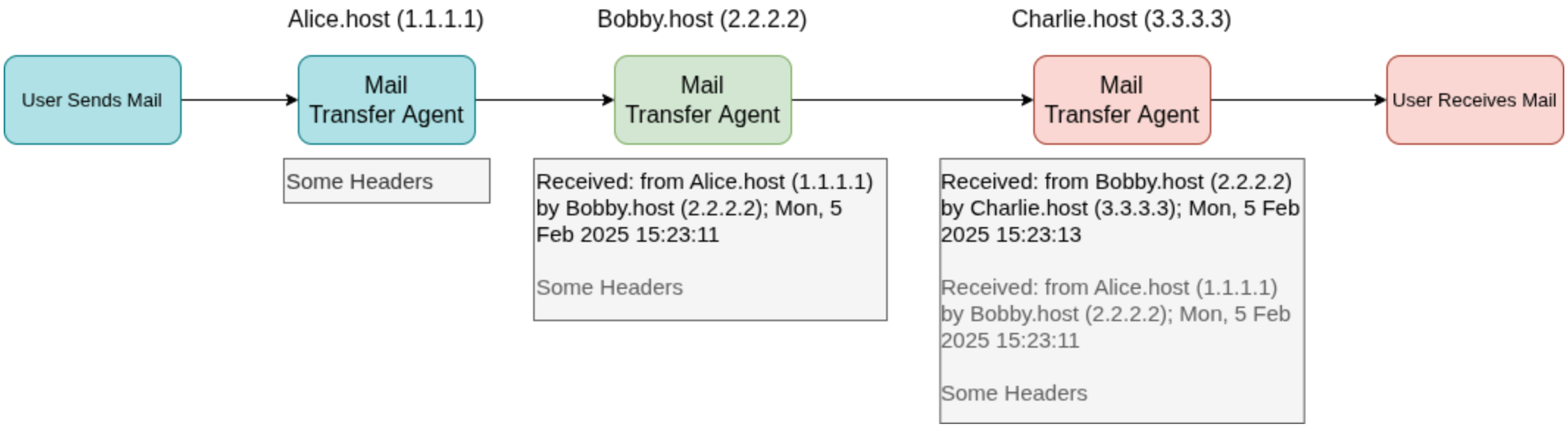
RFC 5321 SMTP October 2008

3.7.2. Received Lines in Gatewaying

When forwarding a message into or out of the Internet environment, a gateway MUST prepend a Received: line, but it MUST NOT alter in any way a Received: line that is already in the header section.

"Received:" header fields of messages originating from other environments may not conform exactly to this specification. However, the most important use of Received: lines is for debugging mail

How to make sense out of Mail relays



```
Received: from 2007-SWREX05.ad.ms.gov.pl (10.11.192.165) by
 2007-SWREX17.ad.ms.gov.pl (10.11.224.165) with Microsoft SMTP Server
 (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id
 15.2.1544.36 via Mailbox Transport; Wed, 26 Nov 2025 03:58:09 +0100
Received: from Pickup by 2007-SWREX05.ad.ms.gov.pl with Microsoft SMTP Server
 id 15.2.1544.36; Wed, 26 Nov 2025 02:58:07 +0000
Received: from 2007-SWREX08.ad.ms.gov.pl (10.11.192.168) by
 2007-SWREX05.ad.ms.gov.pl (10.11.192.165) with Microsoft SMTP Server
 (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id
 15.2.1544.36; Wed, 26 Nov 2025 03:57:02 +0100
Received: from 2007-SWREX08.ad.ms.gov.pl (10.11.192.168) by
 2007-SWREX08.ad.ms.gov.pl (10.11.192.168) with Microsoft SMTP Server
 (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id
 15.2.1544.36; Wed, 26 Nov 2025 03:56:59 +0100
Received: from mail3.wroclaw.sa.gov.pl (10.11.92.132) by
 2007-SWREX08.ad.ms.gov.pl (10.11.192.168) with Microsoft SMTP Server
 (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id
 15.2.1544.36 via Frontend Transport; Wed, 26 Nov 2025 03:56:59 +0100
X-TM-AS-ERS: 5.159.232.10-127.5.252.7
X-TM-AS-SMTP: 1.0 bXgubW9yc2UuaHU= c2VjcmV0YXJpYUBlbWJhY3ViYS5odQ==
X-TM-DDEI-Authentication-Results: spf=pass (sender IP address: 5.159.232.10)
        smtp.mailfrom=embacuba.hu;dkim=pass (signatures verified) header.d=embacuba.
        hu;dmarc=pass p=none header.from=embacuba.hu;
X-DDEI-TLS-USAGE: Used
Received: from mx.morse.hu (unknown [5.159.232.10])
        by mail3.wroclaw.sa.gov.pl (Postfix) with ESMTPS
        for <oddzial.gospodarczy@olsztyn.so.gov.pl>; Wed, 26 Nov 2025 03:56:10 +0100 (CET)
```

Hops

Hop	From	By	With	Date (UTC)	Delay
1	webmail.morse.hu, 192.168.128.216, embacuba.hu, web.dmz.studio64.hu	mx.morse.hu	esmtpa id d01a9121f23	2025-11-26T02:08:25Z	N/A
2	mx.morse.hu, 127.0.0.1	mx.morse.hu, 127.0.0.1	esmtp id 7d2j0x1g-n2n	2025-11-26T02:08:32Z	a few seconds
3	127.0.0.1	mx.morse.hu	esmtp id e97e3122cb2	2025-11-26T02:08:35Z	a few seconds
4	5.159.232.10, mx.morse.hu	mail3.wroclaw.sa.gov.pl	esmtps	2025-11-26T02:56:10Z	an hour
5	mail3.wroclaw.sa.gov.pl, 10.11.92.132	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36 via frontend transport	2025-11-26T02:56:59Z	a minute
6	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36	2025-11-26T02:56:59Z	N/A
7	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	2007-swrex05.ad.ms.gov.pl, 10.11.192.165	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36	2025-11-26T02:57:02Z	a few seconds
8		2007-swrex05.ad.ms.gov.pl	microsoft smtp server id 15.2.1544.36	2025-11-26T02:58:07Z	a minute
9	2007-swrex05.ad.ms.gov.pl, 10.11.192.165	10.11.224.165, 2007-swrex17.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36 via mailbox transport	2025-11-26T02:58:09Z	a few seconds

Hops

Hop	From	By	With	Date (UTC)	Delay
1	webmail.morse.hu, 192.168.128.216, embacuba.hu, web.dmz.studio64.hu	mx.morse.hu	esmtpa id d01a9121f23	2025-11-26T02:08:25Z	N/A
2	mx.morse.hu, 127.0.0.1	mx.morse.hu, 127.0.0.1	esmtp id 7d2j0x1g-n2n	2025-11-26T02:08:32Z	a few seconds
3	127.0.0.1	mx.morse.hu	esmtp id e97e3122cb2	2025-11-26T02:08:35Z	a few seconds
4	5.159.232.10, mx.morse.hu	mail3.wroclaw.sa.gov.pl	esmtps	2025-11-26T02:56:10Z	an hour
5	mail3.wroclaw.sa.gov.pl, 10.11.92.132	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36 via frontend transport	2025-11-26T02:56:59Z	a minute
6	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36	2025-11-26T02:56:59Z	N/A
7	10.11.192.168, 2007-swrex08.ad.ms.gov.pl	2007-swrex05.ad.ms.gov.pl, 10.11.192.165	microsoft smtp server (version=tls gcm_sha384) id 15.2.1544.36	2025-11-26T02:57:02Z	a few seconds
8		2007-swrex05.ad.ms.gov.pl	microsoft smtp server id 15.2.1544.36	2025-11-26T02:58:07Z	a minute
9	2007-swrex05.ad.ms.gov.pl, 10.11.192.165	10.11.224.165, 2007-swrex17.ad.ms.gov.pl	microsoft smtp server (version=tls1_2, cipher=tls_ecdhe_rsa_with_aes_256_gcm_sha384) id 15.2.1544.36 via mailbox transport	2025-11-26T02:58:09Z	a few seconds

Organisation A

Organisation B

Envelope vs Header disambiguation

- *(SMTP)* **Envelope From**
  - Used for message bounces
  - Not visible to end user
  - Might be completely dropped at MTA
- *(Message)* **Header From**
  - Visible in message view
  - More prone to being spoofed

```
..........3D....
```
```
...........220 bzcat.eu ESMTP ready
ehlo [127.0.0.1]
250-bzcat.eu
250-PIPELINING
250-SIZE 50000000
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail FROM:<user@bzcat.eu> size=273
```
Envelope
```
250 2.0.0 OK
rcpt TO:<admin@bzcat.eu>
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
Content-Type: text/plain; charset="utf-8"
Content-Transfer-Encoding: 7bit
MIME-Version: 1.0
Subject: Great Message
From: administrator@company.co
```
Header
```
To: admin@bzcat.eu
Message-ID: <176778568458.1469835.18049631710153193230.message@bzcat.eu>
```

☕ Challenges + Coffee Break

Feel free to ask if you have any questions

We'll continue the presentation at 11:10

Email Authentication Headers

Problems:

- Email is a piece of ancient technology and has many shortcomings in terms of security.
- No security built in by design
- Lots of incentives to abuse it by malicious actors for:
  - Phishing credentials
  - Distributing malware
  - Doing scams

Email Authentication Headers

Solution:

- Sender Policy Framework - **SPF**
- DomainKeys Identified Mail - **DKIM**
- Domain-Based Message Authentication, Reporting and Conformance - **DMARC**

SPF

- Verify if a host can use specific **Envelope From**
- Uses "mechanisms" syntax to specify which hosts are allowed to send emails for given domain
- Stored as a TXT record
- **RFC 7208**

SPF - Example - `cert.pl`

SPF: correct configuration

Domain	`cert.pl`
Record	`v=spf1 include:_spf.cert.pl -all`
Warnings	none
Errors	none

SPF - Example - `cert.pl`

SPF: correct configuration

Domain	`_spf.cert.pl`
Record	`v=spf1 ip4:195.187.55.89 ip4:195.187.240.78 ip4:195.187.242.196 ip4:195.187.242.198 ip4:195.187.6.20 ip4:195.187.6.28 ip4:195.187.242.205 ip4:195.187.5.120 ip4:195.187.6.5 ip4:195.187.6.6 ip4:194.181.250.244 ip4:194.181.250.245 ip4:194.181.250.246 -all`
Warnings	none
Errors	none

SPF - Mechanisms

- `-` Fail - Reject
- `~` Softfail - Recomend reject
- `+` Pass - Accept
- `?` Neutral - Do nothing

RFC 7208 - Sender Policy Framework

DKIM

- Verify message integrity using PKI
- Header includes:
  - A list of headers to be signed
  - Domain and selector used to retrieve the public key
  - Message Checksum
  - Others

DKIM - Example

dkim-signature

v=1; a=rsa-sha256; q=dns/txt; c=relaxed/relaxed; d=dh-lines.com; s=default; h=Content-Type:MIME-Version:Message-ID:Date: Subject:To:From:Reply-To:Sender:Cc:Content-Transfer-Encoding:Content-ID: Content-Description:Resent-Date:Resent-From:Resent-Sender:Resent-To:Resent-Cc :Resent-Message-ID:In-Reply-To:References:List-Id:List-Help:List-Unsubscribe: List-Subscribe:List-Post:List-Owner:List-Archive; bh=AnQJhdiza1K4KIyJSaRZhT+GrABtctfwaE5hYMsS1xc=; b=jC0mRQC5kOOpZb+6gmdmVKuur5 OBc0ccO9N8ovsJHc+AuiwiYX3VDLyvAYkSGxU9/XPfcCE6wz7dqdcQuBXzRkHUfBQ+urcBUbsCW2R 7mWB0/DRersGnwqhONoQBM30n43xJ+FXEYVeX7V6kvaarjf1dBCF7Yzm1ddLoZHgcoVvbdmNumhWnQHJYwdWXRlgmHnfV8ZORbamqEE3ee9/Kz+WZDn8FFFTjHPp2SbYQ/fixMyqL1jqYB2MWHR+jpfr8R TEkFtjUx6BIC2XO6RWT4xUyzBKyTWtXbomZzlsBm77cqrUS3yK5AY5kkYWGHCESjjxU7byCC/niO1 HudZP6qQ==;

- `d=dh-lines.com` - domain to be queried for public key
- `s=default` - selector specifying which key (subdomain) should be used
- `h=Content-Type:MIME-Version:...` - list of headers to be hashed
- `bh=AnQ...` - hash of attached bodies
- `b=jC0m...` - calculated siganture

DKIM - Retrieving public key

- TXT Record format: `{selector}._domainkey.{domain}`

```
>> dig default._domainkey.dh-lines.com -t TXT +short

"v=DKIM1; k=rsa; p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ
8AMIIBCgKCAQEAqsCxHwvceXhBYOg6oz/eKsG5a5nejMPDd68
TJuYx9C8/nEvCyyLFyHSRwakMIlN/ExteA9hlogBAaY6rU0cF
LSdFQhOoIr73kdBgZMWnQ5JaAMpvzu8JRtr044cRqQnOOEEe3
H1Ecl8uCPfClKuyceOyBwfLAtqFkDHdabm4VX2dEpcRJBdTrQ
R6qTvwIWYj1" "rbCJ7OSbKjdhpqYZ0ARVge+mriDwwe1UL/T
45hEuHKwZbk3VRMUmy5pudyxdoH1a/7tq/sM1lZmYl4JDSR70
TX02CIwGJTQmSI1CLQNVBe9k8mEtxD5tGBxSIdmrHbwHMOJRx
CFOSb0FWolv9JuQQIDAQAB;"
```

DMARC

- SPF
  - Verifies `Envelope From` validity
- DKIM
  - Verifies message integrity using a public key stored in an arbitrary domain record.
- Neither one of them actually fully verifies the `From` Header
- This is what DMARC is used for

DMARC

- TXT record stored in `_dmarc.{domain}`
- Instructs the receiving mail server what should happen to an email message that doesn't pass the authentication check
- Verifies that domains verified by SPF/DKIM match `Header From`

```
DMARC authentication pass =
(SPF authentication pass AND SPF identifier alignment)
OR
(DKIM authentication pass AND DKIM identifier alignment)
```

DMARC - Example

```
dig _dmarc.cert.pl -t TXT +short
"v=DMARC1; p=reject; rua=mailto:dmarc-cert@cert.pl; ruf=mailto:security@cert.pl; fo=s"
```

⚠ DMARC: configuration warnings

Domain	`cert.pl`
Record	`v=DMARC1; p=reject; rua=mailto:dmarc-cert@cert.pl; ruf=mailto:security@cert.pl; fo=s`
Warnings	Using the ruf tag is not recommended, as it's not supported by multiple e-mail providers.
Errors	none

DMARC - Example

- `p=reject` - what action should be taken if the check fails
  - `none` - do nothing, just report
  - `quarantine` - recommend moving the message to spam
  - `reject` - recommend rejecting the message
- `rua=mailto:dmarc-cert@cert.pl` - email address used for reporting abuse by mail servers
- `aspf=r` - SPF alignment restriction (`strict` / `relaxed`)
- `adkim=r` - DKIM alignment restriction (`strict` / `relaxed`)

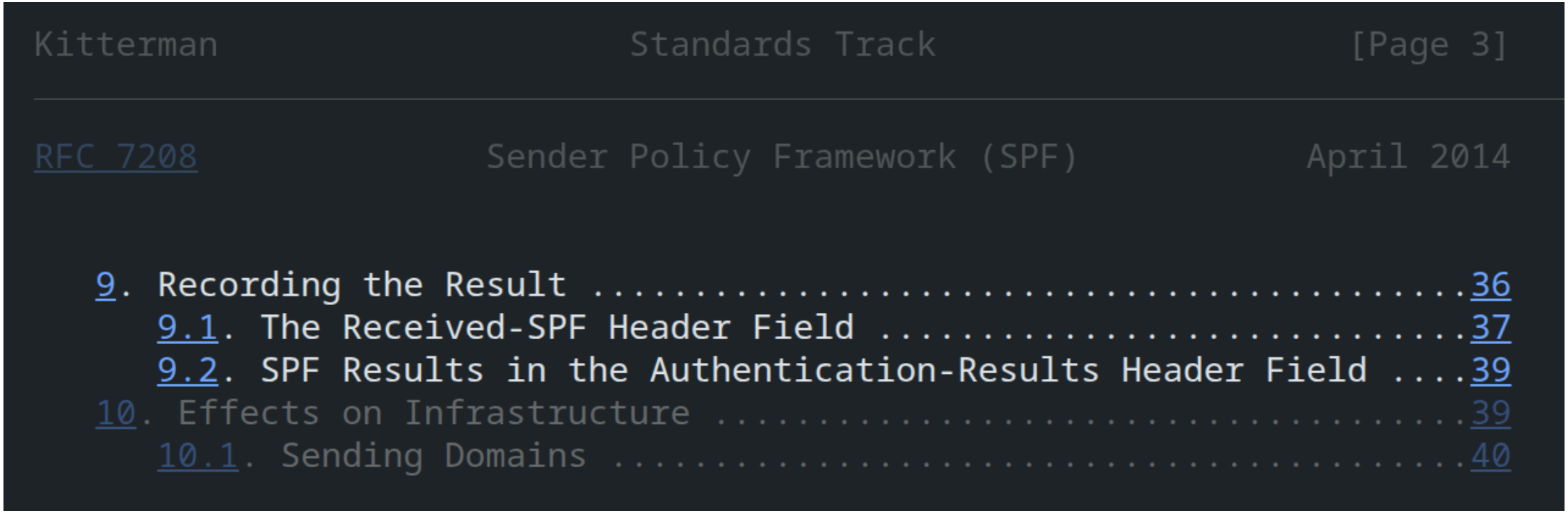
Bonus: Authenticated Received Chain

- A way of chaining message authentication information across relays
- Each relay signs its authentication result header using PKI
- **RFC 8617 - The Authenticated Received Chain (ARC) Protocol**
- Still pretty new and not used everywhere

Email abuse

How to triage spoofed emails - SPF/DKIM/DMARC

Spotting SPF fails



Kitterman Standards Track [Page 3]

RFC 7208 Sender Policy Framework (SPF) April 2014

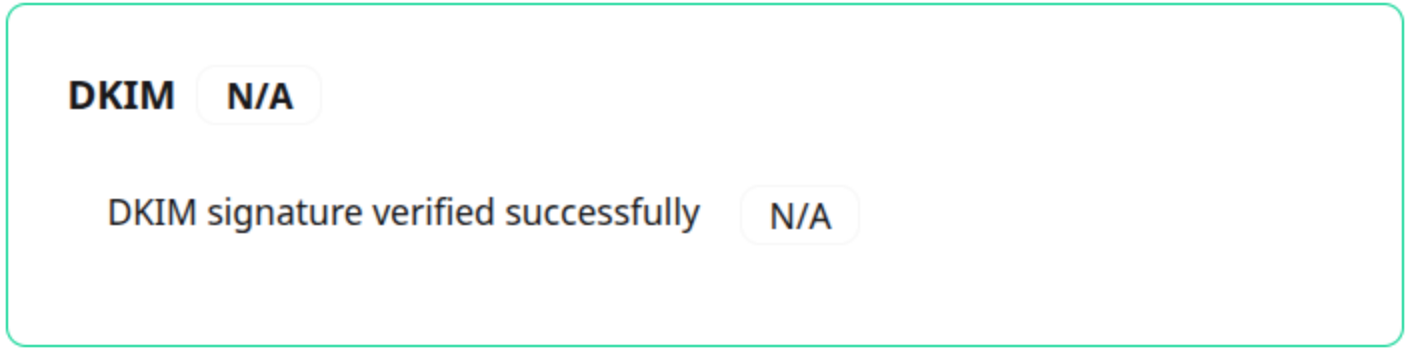
RFC 7208 - Sender Policy Framework

Spotting SPF fails - examples

- ✅ pass
- ❌ fail

Spotting DKIM fails

Verdicts




```
>> pipx install dkimpy
>> cat 0192db34ecd5a05a357651d09ec29227f54a01687da9543e74e9e74a98f04699 | dkimverify
signature ok
```

Spotting DKIM fails - examples

- ✅ pass
- ❌ fail

How to triage spoofed emails - no security

- It's hard
- Reverse IP/Passive DNS lookups
- Look at other headers

Can you send an email originating from a NX domain

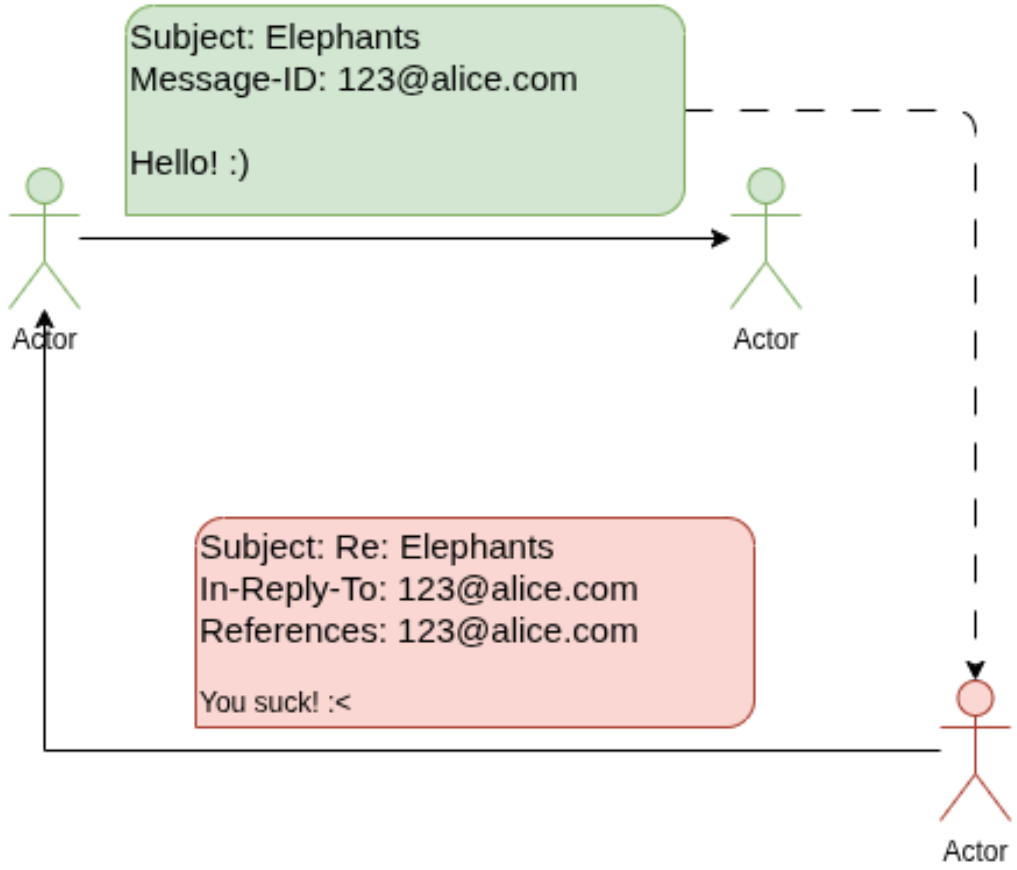
```
220 bzcat.eu ESMTP ready
ehlo [127.0.0.1]
250-bzcat.eu
250-PIPELINING
250-SIZE 50000000
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail FROM:<alice@this-address-does-not-exist.com> size=213
250 2.0.0 OK
rcpt TO:<admin@bzcat.eu>
450 4.1.8 <alice@this-address-does-not-exist.com>: Sender address rejected: Domain not found
rset
250 2.0.0 Ok
```

Can you send an email originating from a domain without MX record

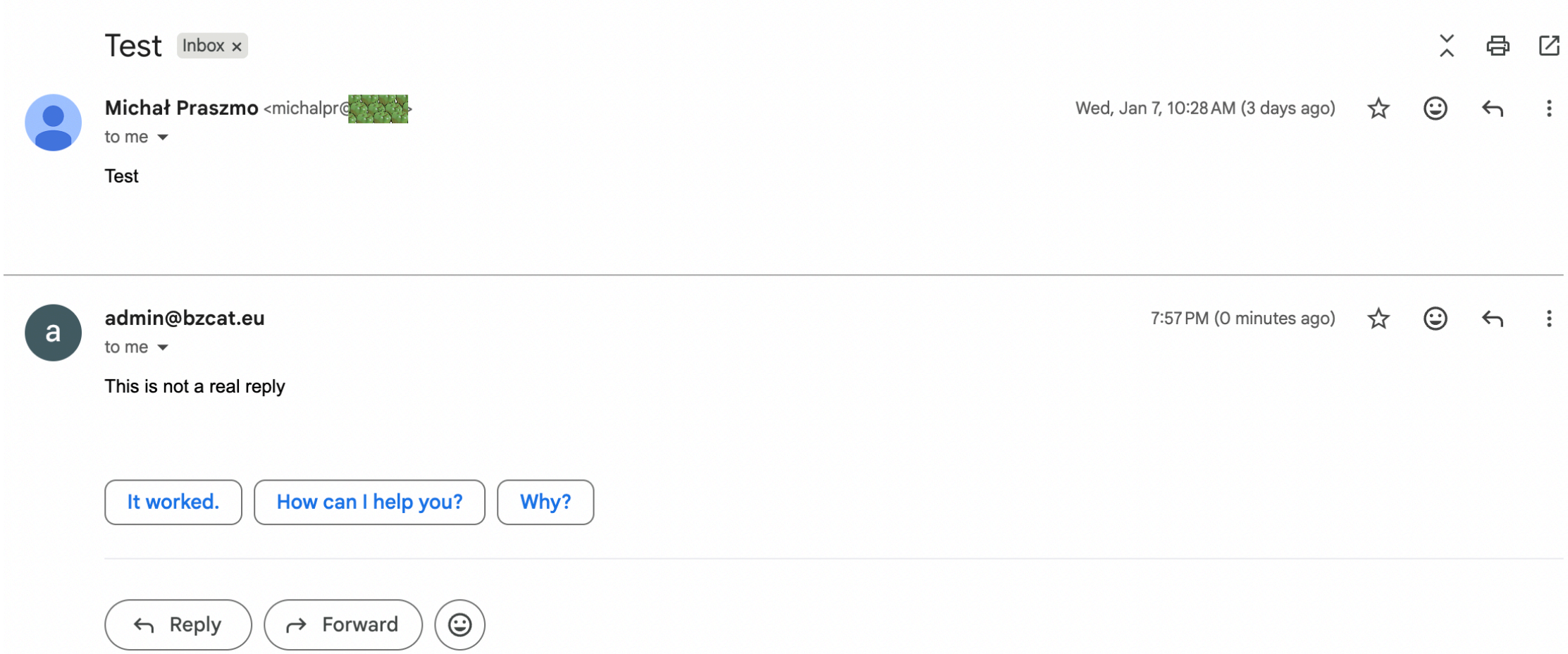
```
220 bzcat.eu ESMTP ready
ehlo [127.0.0.1]
250-bzcat.eu
250-PIPELINING
250-SIZE 50000000
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail FROM:<alice@example.com> size=193
250 2.0.0 OK
rcpt TO:<admin@bzcat.eu>
550 5.7.27 <alice@example.com>: Sender address rejected: Domain example.com does not accept mail (nullMX)
rset
250 2.0.0 Ok
```

Conversation hijacking - how does it work?

- Attacker gains access to the sent email
- Sets correct headers:
  - `In-Reply-To`
  - `References`
  - `Subject`



Conversation hijacking - example



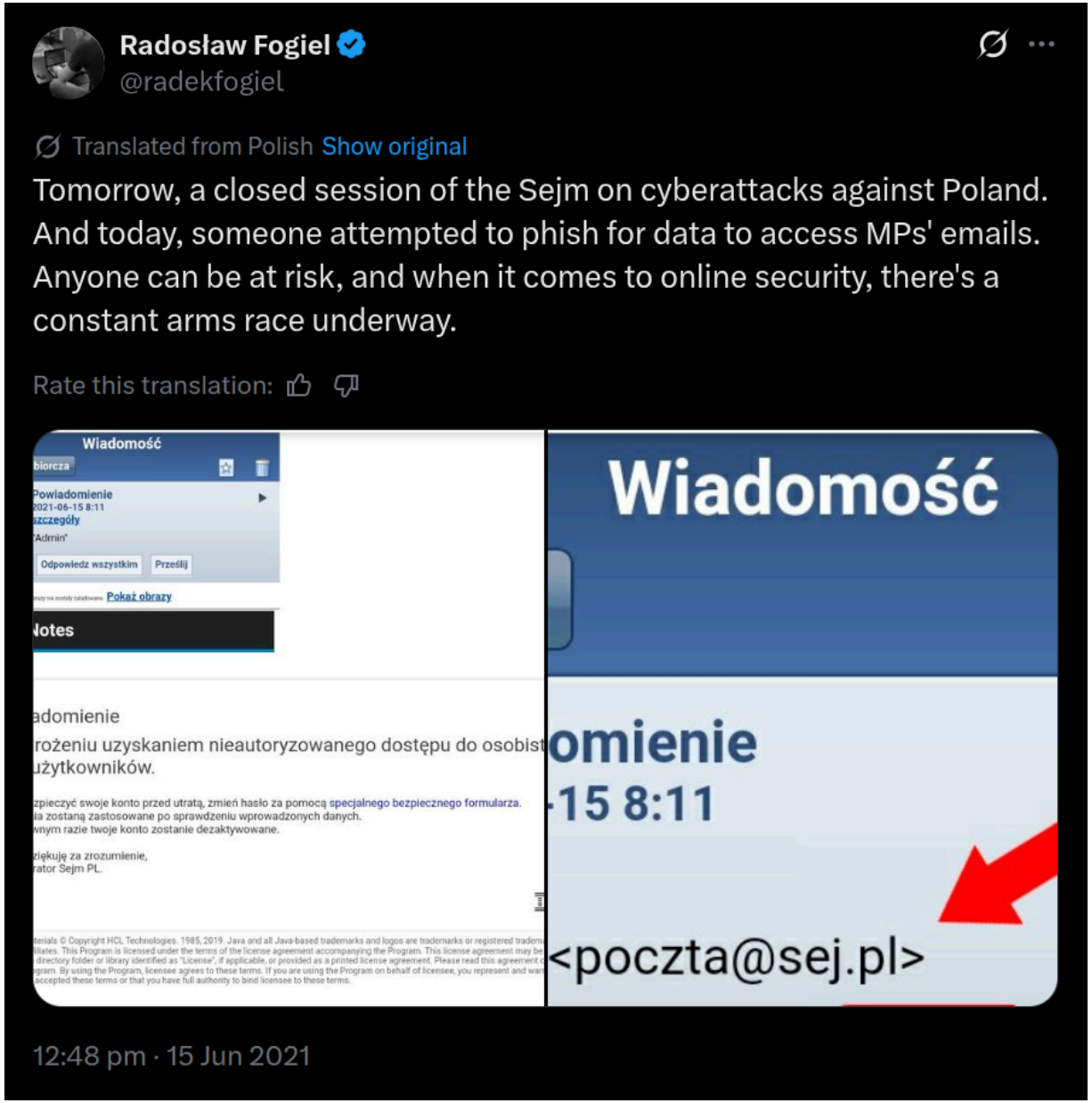
Typosquatting

- pretty straightforward?
- register a domain that mimics the original one
- send an email to the victim

Typosquatting

it's not straightforward

- `sejm.pl` - legitimate
  - ✅ Good SPF record
  - ✅ Good DMARC record
- `sej.pl` - spoofed
  - 😕 SPF with `softfail`
  - ❌ No DMARC record



Typosquatting

it's not straightforward

~~Typosquatting~~

Accidental Typosquatting

~~Typosquatting~~

~~Accidental Typosquatting~~

Typos

- Both users and administrators make mistakes
- Attackers may use that to passively collect confidential information
- Very hard to combat in practice
  - Levenshtein distance
  - Fat Finger distance

psse.sanok@pis.gov.pl	tel. 13 464 12 19 fax 13 464 12 19
psse.stalowawola@pis.gow.pl	tel. 15 842 51 30 fax 15 842 11 81
psse.strzyzow@pis.gov.pl	tel./fax 17 276 11 52 tel. dyr. 17 276 14 03

Adresy e-mail do kontaktu:
koordynacjaswiadczeneu@katowice.uw.gow.pl
koordynacjaeu@katowice.uw.gow.pl

1. Розпорядником персональних даних є Воєвода Лодзький. Резиденція Воєводи Лодзького є Лодзинське Воєводське Управління в Лодзі вул. Пьотрковська 104, 90-926 м. Лодзь. Звязатись можна за телефоном: /42/ 664-10-00; адреса електронної пошти: kancelaria@lodz.uw.gov.pl, skrytki ePUAP: /lodzuw/skrytka.
2. В питаннях пов'язаних з персональними днями можна звертатись до інспектора охорони персональних даних за адресою електорнної пошти: iod@lodz.uw.gow.pl

złożenie wniosku do kierownika biura powiatowego Agencji Restrukturyzacji i Modernizacji Rolnictwa właściwego ze względu na miejsce zamieszkania lub siedzibę wnioskodawcy. Formularz wniosku wraz z instrukcją wypełniania jest dostępny na stronie ARiMR
http://www.arimr.gow.pl/dla-beneficjenta/wnioski/ewidencja-producentow.html

XSS exploitation - CVE-2024-42009

cert.pl - Roundcube XSS analysis

> UNC1151 campaign exploiting a vulnerability in Roundcube software to steal credentials _

June 5, 2025 | CERT Polska | #analysis , #CVE-2024-42009 , #roundcube , #unc1151

This week, the CERT Polska team detected an email campaign targeting Polish entities that exploits the CVE-2024-42009 vulnerability in Roundcube software. The vulnerability allows JavaScript code to be executed upon reading an email, which has been used in credential-stealing attacks.

It's worth noting that another vulnerability, CVE-2025-49113 , was also published this week, allowing remote code execution on a server by an authenticated email user. While we haven't yet seen it exploited, combining these two vectors could lead to a complete takeover of the targeted organization's infrastructure.

Technical indicators allow us to link these activities with a high degree of probability to the UNC1151 cluster. According to publications by Mandiant and Google , the UNC1151 group is linked to the Belarusian government, and according to other sources, it also has ties to Russian intelligence agencies.

XSS exploitation - CVE-2024-42009

- Roundcube XSS on email open
- Used by an APT against Polish organisations in May of 2025
- CVE-2025-49113 was chained in later campaigns to achieve RCE on the email servers

Wrap up

- Reality is often more complicated
- Examples were picked with simplicity in mind
- Sometimes it's impossible to say what has happened with 100% certainty
- Let us know if you have any ideas for improving EML analyzer 🙂

🔨 Challenges

Thank you for your attention

Feel free to use the remaining time to complete the challenges and explore the examples linked in slides.

Or enjoy an early lunch 😋