Spoofing? - Verfahren zur Absenderprüfung bei E-Mail

- **Spoofing und Absenderprüfung**
- **Wir versenden Mails nur von … (SPF)**
- **… und signiert sind unsere Mails auch (DKIM)**
- **… und man darf sich darauf verlassen (DMARC)**
- **… und da war noch das mit den Maillinglisten (ARC)**

Absenderfälschungen / Spoofing

```
From: <projektleiter@meine-uni.example.org>
To: <beschaffung@meine-uni.example.org>
Subject: [DRINGEND] Kontodaten Firma X geändert
Date: Wed, 29 Jan 2025 15:09:34 +0100

Die ausstehenden Beträge für die Dienstleistungen der
Firma X bitte ab Februar auf Kontonummer
  GB94 BARC 1020 1530 0934 59
überweisen.

Sorry für die kurzfristige Mitteilung!
```

Zum Durchlesen (bzw. von Juristen erklären lassen) dazu: OLG Schleswig 12 U 9/24

Absenderfälschungen

- Versand mit einer Mailadresse, die einem nicht gehört
- Geht systembedingt immer
- Sieht man dann an den Trace-Headern (`Received:`)
- Aber „man“ ≠ „die Person, die solche Mails erhält“

Lösungsversuche

- Naivlösung
  - Meine Maildomains, meine Server
  - ⇒ Mails mit meinen Absenderadressen können nur von meinen Server kommen!
  - ⇒ Mails von externen Servern mit internen Adressen lehnen wir ab.
- Zu vehement
  - Weiterleitungen, Externe Dienstleister?
  - Absenderprüfung außerhalb der Einrichtung?

Spoofing? - Verfahren zur Absenderprüfung bei E-Mail

- **Spoofing und Absenderprüfung**
- **Wir versenden Mails nur von … (SPF)**
- **… und signiert sind unsere Mails auch (DKIM)**
- **… und man darf sich darauf verlassen (DMARC)**
- **… und da war noch das mit den Maillinglisten (ARC)**

So einfach ist das nicht ...

- Mails von „draußen“
  - Weiterleitungen zurück ans Institut
  - Mailinglisten mit mehreren Abonnenten am Institut (Institut → Liste → Institut)
- Erkennbarkeit für andere
  - Woher soll ein externes System wissen, wer für eine Domain E-Mails versenden darf?
- Idee:

  Publiziere im DNS, welche Rechner Mails für eine Domäne versenden
- → Sender Policy Framework (RFC 7208)

Prüfparameter

- <ip>

  Adresse des einliefernden Systems (Remote IP)
- <identity>
  Identität des Absenders (dazu später mehr)
- <domain>
  Zu prüfende Domain. Wird zunächst auf <identity> gesetzt.

Sender Policy Framework

SPF-Identitäten

- „MAIL FROM“ Identity (RFC 7208, Section 2.4)
  - Geprüft wird der Domänenname des Return Path (Envelope Sender)
  - Bei leerem Return-Path wird `postmaster@${HELO}` angenommen

    `MAIL FROM: <postmaster@example.org>` → `example.org`

    `MAIL FROM: <root@rechner1.example.org>` → `rechner1.example.org`
- „HELO“ Identity (RFC 7208, Section 2.3)
  - Zusätzliche Prüfung neben „MAIL FROM“
  - Besonders bei leerem MAIL FROM:
    - ```
      EHLO rechner1.example.org
      MAIL FROM: <>
      → rechner1.example.org
      ```
      ```

Sender Policy Framework

SPF-Identitäten

- „MAIL FROM“ Identity (R
  - Geprüft wird der D
  - Bei leerem Retu

    `MAIL FROM: <p`

    `MAIL FROM: <roo`
- „HELO“ Identity (RFC 7208,
  - Zusätzliche Prüfung neb
  - Besonders bei leerem MAIL FROM:
    - `EHLO rechner1.example.org`
      `MAIL FROM: <>`
      → `rechner1.example.org`

- RFC 7280 sagt:
- EHLO soll zuerst geprüft werden. Bei eindeutigem Resultat kann unter Umständen die Prüfung von
- MAIL FROM: verzichtet werden.

Praktische Umsetzung (postfix-policyd-spf-perl):

- Wenn die HELO-Prüfung ein negatives Resultat liefert → negativ
- Bei leerem MAIL FROM wird das HELO-Resultat übernommen
- Ansonsten wird MAIL FROM geprüft.

Wie wird geprüft?

- Auf Basis des Namens wird im DNS ein passender TXT-Eintrag gesucht, der mit `v=spf1` beginnt:

  "meine-uni.example.org IN TXT „v=spf1 …"

- Ein SPF-Record besteht aus einer Liste von Mechanismen
  - Mechanismen werden von links nach rechts ausgewertet
  - Jeder Mechanismus hat eine Bedingung und ein Resultat
  - Trifft die Bedingung zu, wird das Resultat zurückgegeben

Wie wird geprüft?

Mögliche Resultate

- + (pass): Prüfung erfolgreich
- - (fail): Prüfung fehlgeschlagen
- ~ (softfail): „Vermutlich Fehlschlag“
- ? (neutral): Keine Aussage

- + ist für Mechanismen optional
- Standardresultat ist „neutral“

Wie wird geprüft?

```
v=spf1 +A MX include:extern.example.org -all
```

- +A (= A)
  Wenn die DNS-Auflösung (A oder AAAA) der <domain> zurückliefert <ip> zurückliefert
  - auch: A:a.example.org (wenn die Auflösung von a.example.org <ip> zurückliefert)
- MX (= +MX)
- Wenn <domain> einer der MXe der <domain> ist
  - auch: MX:other.example.org (wenn <ip> ein MX von other.example.org ist)

Wie wird geprüft?

```
v=spf1 +A MX include:extern.example.org -all
```

- include:extern.example.org
  - Setze <domain> auf extern.example.org und werte den Record dort aus
  - Es wird **nicht** das Resultat der Einbindung übernommen. Stattdessen:
    - „pass“ vom eingebundenen Record → match
    - „fail/softfail/neutral“ → not match
  - Praktisch für externe Versender (include:_spf.newsletterversand.example.org) aber auch zur internen Strukturierung (include:spf.meine-uni.example.org)

SPF Macros

Dynamische Expansion im SPF-Record

```
v=spf1 exists:%{d}
```

- %{ir}: reverse <ip> (wie bei Reverse DNS Zones)
- %{v}: „in-addr“ für IPv4, „ip6“ für IPv6
- %{l}: Lokalteil der MAIL FROM-Adresse
- %{o}: Domäne der MAIL FROM-Adresse

Absenderfälschungen / Spoofing

```
From: <projektleiter@meine-uni.example.org>
To: <beschaffung@meine-uni.example.org>
Subject: [DRINGEND] Kontodaten Firma X geändert
Date: Wed, 29 Jan 2025 15:09:34 +0100

Die ausstehenden Beträge für die Dienstleistu
Firma X bitte ab Februar auf Kontonummer
  GB94 BARC 1020 1530 0934 59
überweisen.

Sorry für die kurzfristige Mitteilung!
```

Für diese Mail kann eine gültige SPF-Validierung mit geringem Aufwand herbeigeführt werden

Bringt das wirklich was?

- Kein Problem mit Mailinglisten
- Der Return Path wird dort ohnehin umgeschrieben
  - Sympa: listname-owner@ Mailman: listname-bounces@
  - Beides zusätzlich mit VERP-Subadressen

- MAIL FROM
  - Als Spamschutz wenig hilfreich
  - Als Spoofing-Schutz auch nicht
- Prüft einspeisenden Rechner

Spoofing? - Verfahren zur Absenderprüfung bei E-Mail

- **Spoofing und Absenderprüfung**
- **Wir versenden Mails nur von … (SPF)**
- **… und signiert sind unsere Mails auch (DKIM)**
- **… und man darf sich darauf verlassen (DMARC)**
- **… und da war noch das mit den Maillinglisten (ARC)**

… und signiert sind unsere Mails auch (DKIM)

Können wir die Mails nicht digital signieren?

- Idee:
  - Wir signieren ausgehende Nachrichten
  - Öffentlicher Schlüssel liegt im DNS
  - Jeder kann die Signatur prüfen
- → DomainKeys Identified Mail
  - RFC 4686: Analysis of Threats Motivating DomainKeys Identified Mail, RFC 5585: DKIM Service Overview, RFC 5617: DKIM Author Domain Signing Practices, RFC 5863: DKIM Development, Deployment and Operations, RFC 6376: DKIM, RFC 6377: DKIM and Mailing Lists
  - … langt dann auch.

Was wird signiert?

- Header
  - Simple:
    Header werden so signiert wie sie in der Mail stehen
    → Jede Änderung (auch Folding) im Transport bricht die Signatur
  - Relaxed:
    Lower Case, Unfolding, Space Folding, Trailing Whitespace
- Body
  - Simple:

    Leerzeilen am Ende des Nachrichteninhalts werden ignoriert
  - Relaxed:

    Wie Simple plus:
    - Trailing Whitespace
    - Space Folding
  - Muss: CR | LF → CRLF
  - Kann: 8bitmime (Content-Transfer-Encoding)

Was wird signiert?

- Header
  - Simple:
    Header werden so sign
    der Mail stehen
    → Jede Änderun
    Transport
  - Relaxed:
    Lower Case, Unfoldi
    Folding, Trailing Whit

achrichteninhalts

ransfer-

Etwas Statistik:

relaxed/relaxed	63,96 %
relaxed/simple	26,85 %
simple/relaxed	0,06 %
simple/simple	9,13 %

(eingehende Mails an der CAU
01.01.2025 bis 13.05.2025)

Wie wird signiert?

- RSA
  - Schlüssellänge
  - BSI TR-03182:
    $1024 \leq bits(p) \leq 2048$
  - Für 4096 RSA-Schlüssel braucht man schon EDNS0
  - Auch ein 2048-Bit RSA-Schlüssel benötigt bereits mehrere TXT-Strings

- ED25519
  - Kürzere Schlüssel (512 Bit)
  - Passt in einen einzigen TXT-String

Elliptische Kurven

- RFC 8463 definiert ED25519-SHA256 für DKIM-Signaturen
- Die sind schön kurz:

```
20250116ed25519._domainkey.uni-kiel.de. TXT "v=DKIM1; k=ed25519; p=WXYapoFcTdW0GjTLrjTUZgMYDYfcUwMrKs++rbueq8Q="
```

- BSI-TR-03182
  - Double Signing (RSA und ED25519) ist Pflicht
  - Mögliche Fehlkonfiguration: Nur prüfen, ob irgendwo „fail“ vorkommt und „pass“-Ergebnis für zweite Signatur ignorieren
  - Ein System, dass kein ED25519 kann, ist nicht konform
  - Aber: Immer noch oft nicht unterstützt (Hey, Google!)

Elliptische Kurven

- Am Rande
  - Bei ED25519 steht die Schlüssellänge fest
  - Im DNS wird der Schlüssel ohne ASN.1-DER-Header base64-kodiert hinterlegt

```
openssl genpkey -algorithm ed25519 -outform DER \
| openssl asn1parse -inform DER
    0:d=0  hl=2 l=  46 cons: SEQUENCE
    2:d=1  hl=2 l=   1 prim: INTEGER           :00
    5:d=1  hl=2 l=   5 cons: SEQUENCE
    7:d=2  hl=2 l=   3 prim: OBJECT            :ED25519
   12:d=1  hl=2 l=  34 prim: OCTET STRING      [HEX DUMP]:04208E396697404C7E7….]
```

Verweis auf extern

- _domainkey kann auch ein CNAME sein
- … in eine andere Domain

```
mwo._domainkey.uni-kiel.de.     IN CNAME mwo.domainkey.u29988855.wl106.sendgrid.net.
mwn._domainkey.uni-kiel.de.     IN CNAME mwn.domainkey.u29960932.wl183.sendgrid.net.
mwo2._domainkey.uni-kiel.de.    IN CNAME mwo2.domainkey.u29988855.wl106.sendgrid.net.
mwn2._domainkey.uni-kiel.de.    IN CNAME mwn2.domainkey.u29960932.wl183.sendgrid.net.
```

Verweis auf extern

- _domainkey kann auch ein CNAME sein
- … in eine andere Domain

```
mwo._domainkey.uni-ki                    IN CNAME mwo.domainkey.u29988855.wl106.sendgrid.net.
mwn._                                                                    id.net.
mwo2.                                                                    rid.net.
mwn2.                                                                    rid.net.
```

Granularität

- Bei SPF ist es möglich, Regeln auch für einzelne Adressen vorzugeben
- Ein DKIM-Selector gilt immer für die gesamte Domäne
- Wollte man so, ist halt jetzt so.

DKIM-Signature-Header

```
DKIM-Signature: v=1; a=ed25519-sha256; q=dns/txt; c=relaxed/relaxed;
        d=uni-kiel.de; s=20250116ed25519; h=Subject:From:To:MIME-Version:Date:
        Message-ID:Content-Type:From:Reply-To:Subject:Cc:In-Reply-To:References:
        Content-Transfer-Encoding:Content-ID:Content-Description;
        bh=JvZuaxa+cz/78sUdjgXppxSjCmc6An6OGW3YJdAmaLA=; i=@uv.uni-kiel.de; b=cd4C/nk
        XnRNi6C3hF4LSdZYzm21mSHCAwU9xRNsmPAz69yOz8MGGYmVt3xbIh4IqWmchm4AnoZ0gN3P2S2M+
        AA==;
```

DKIM-Signature-Header

```
DKIM-Signature: v=1; a=ed25519-sha2                    d/relaxed;
        d=uni-kiel.de; s=                              -Version:Date:
        Message-ID:Conte                               o:References:
        Content-Transfer
        bh=JvZuaxa+cz/78                               iel.de; b=cd4C/nk
        XnRNi6C3hF4LSdZYzM                             hchm4AnoZ0gN3P2S2M+
        AA==;
```

Signing Domain Identifier

Wer ist für die Signatur "verantwortlich"

→ In welcher Domäne liegt der TXT-Record?

DKIM-Signature-Header

```
DKIM-Signature: v=1; a=ed25519-sha256; q=dns/txt; c=relaxed/relaxed;
        d=uni-kiel.de; s=20250116ed25519; h=Subject:From:To:MIME-Version:Date:
        Message-ID:Co                                  To:Subject:Cc:In-Reply-To:References:
        C                                                          scription;
                                                                  .uni-kiel.de; b=cd4C/nk
                                                                  Wmchm4AnoZ0gN3P2S2M+
```

s=20250116ed25519

Selector

→ Welcher Schlüssel wird benutzt?
→ Public Key liegt unter

20250116ed25519._domainkey.uni-kiel.de IN TXT

Achtung: Die Versuchung ist groß, aber der Selektor darf offiziell keine Unterstriche enthalten

DKIM-Key-Rollover

- DKIM-Schlüssel sollen regelmäßig getauscht werden
- BSI empfiehlt:
  mindestens alle drei Monate

  → machen alle (auch teilweise große Anbieter) viel zu selten
- Für ganz Mutige:
  alte private Schlüssel nach $Zeit durch Veröffentlichung unbrauchbar machen

DKIM-Signature-Header

```
DKIM-Signature: v=1; a=ed25519-sha256; q=dns/txt; c=relaxed/relaxed;
        d=uni-kiel.de; s=20250116ed25519; h=Subject:From:To:MIME-Version:Date:
        Message-ID:Content-Type:From:Reply-To:Subject:Cc:In-Reply-To:References:
        Content-Transfer-Encoding:Content-ID:Content-Description;
        bh=JvZuaxa+cz/78sUd[illegible]mc6An6OGW3YJdAmaLA=; i=@uv.uni-kiel.de; b=cd4C/nk
```

Von der Signatur erfasste Header

- In der Reihenfolge wie angegeben von **unten nach oben**
- Bei mehrfacher Erwähnung wird der jeweils n-te Header mit signiert
- Gibt es einen zu signierenden Header in der Nachricht nicht, wird er als Leerstring angenommen (Oversigning)

DKIM-Signature-Header

Agent or User Identifier

- [<local_part>]@<domain>
- Ausgewertet wird nur der Domänenteil
- i= muss d= oder einer Subdomäne von d= entsprechen

- **DKIM verlangt ausdrücklich keinen Zusammenhang zwischen Headern (From:) und i=**

```
laxed/relaxed;
m:To:MIME-Version:Date:
c:In-Reply-To:References:
scription;
; i=@uv.uni-kiel.de; b=cd4C/nk
l3xbIh4IqWmchm4AnoZ0gN3P2S2M+
```

DKIM-Signature-Header

```
DKIM-Signature: v=1; a=ed25519-sha256; q=dns/txt; c=relaxed/relaxed;
        d=uni-kiel.de; s=20250116ed25519; h=Subject:From:To:MIME-Version:Date:
        Message-ID:Content-Type:From:Reply-To:Subject:Cc:In-Reply-To:References:
        Content-Transfer-Encoding:Content-ID:Content-Description;
        bh=JvZuaxa+cz/78sUdjgXppxSjCmc6An6OGW3YJdAmaLA=; i=@uv.uni-kiel.de; b=cd4C/nk
        XnRNi6C3hF4LSdZY                CAwU9xRNsmPAz69yOz8MGGYmVt3xbIh4IqWmchm4AnoZ0gN3P2S2M+
        AA==;
```

Header Hash
Signatur der Header (siehe weiter vorne)

Eigenschaften

- Einfügen/Ändern nicht-mitsignierter Header bricht die Signatur nicht
- Vorne angefügte Header brechen die Signatur nicht
- Außer bei Oversigning (From:From)
- Änderungen an der Nachricht brechen die Signatur

DKIM und Mailinglisten

Eigentlich alles super, aber

- Die meisten Mailinglisten schreiben die Betreffzeile um
  - Ketzerei: Man muss den Betreff nicht mitsignieren (do not do this at home)
- Disclaimer/Unsubscribe am Nachrichtenende
  - Letzteres kann man theoretisch mit Body Length Limit (l=) umgehen
  - RFC 6376 sagt recht deutlich „lass das lieber“

Non-Repudiation

- Seiteneffekt
  - Jede E-Mail wird mit einer kryptographischen starken Signatur versehen
  - … die auch für jeden überprüfbar ist.
  - auch nachträglich, solange der öffentliche Schlüssel verfügbar ist
  - Ziel:
    „Diese Mail wurde nicht von uns versendet“ für Mails, die man nicht versendet hat
  - Seiteneffekt:
    „Diese Mail wurde nicht von uns versendet“ für Mails, die man versendet hat ist dann auch falsifizierbar (non-repudiation)

Wo signieren?

Posteingangserver

- Vorteil
  - Nahe an der Nutzerauthentifizierung
- Nachteil:
  - Danach ist die Nachricht unveränderlich
  - Das kann aber auch Argumentverstärker sein

Postausgangsserver

- Vorteil:
  - Das ist nach allen internen Umbauten
  - Nach dem Listserver
- Nachteil
  - Weit weg von der Nutzerauthentifizierung

Wo signieren?

Posteingangserve

- Vorteil
  - Nahe an der Nutzer
- Nachteil:
  - Danach ist die Nach
    unverändlich
  - Das kann aber au
    Argumentverstärker sein

r

n

Nachteil:

- Weit weg von der Nutzerauthentifizierung

Signiert man rein interne Mails auch?

→ Man bekommt Infrastruktur für die Validierung von Absendern zwischen lose gekoppelten Systemen als Bonus

Christian-Albrechts-Universität zu Kiel

Spoofing? - Verfahren zur Absenderprüfung bei E-Mail

- **Spoofing und Absenderprüfung**
- **Wir versenden Mails nur von … (SPF)**
- **… und signiert sind unsere Mails auch (DKIM)**
- **… und man darf sich darauf verlassen (DMARC)**
- **… und da war noch das mit den Maillinglisten (ARC)**

Domain-based Message Authentication, Reporting and Conformance

Wie informiert man die Welt?

- Wir machen DKIM und/oder SPF (Authentication)
- … und nehmen das **so** ernst (Conformance)
- Bitte informiert uns, wenn etwas nicht stimmt (Reporting)
- Domain-based Message Authentication, Reporting and Conformance (DMARC, RFC 7489)

  → Noch ein TXT-Record

DMARC-Records

```
IN TXT (
  "v=DMARC1; "
  "p=quarantine; "
  "sp=reject; "
  "rua=mailto:dmarc-rua@example.org; ri=86400"
  "ruf=mailto:dmarc-ruf@example.org; "
  "fo=0; "
  "adkim=r; "
  "aspf=s; "
  "pct=50"
)
```

DMARC-Records

```
"adkim=r; "
"aspf=s; "
```

- Bei DMARC muss man SPF und DKIM einsetzen
- Hier spezifiziert: Alignment
  - relaxed
  - strict

DMARC-Alignment

- DMARC ändert und ergänzt SPF und DKIM
  - Bei DMARC wird für SPF der Header-From: überprüft (nicht der Return-Path)
  - DMARC verlangt allerdings Identifier Alignment
    - **strict**
      DKIM: d= muss der From-Domäne entsprechen
      SPF: Return-Path muss in derselben Domäne wie die From:-Adresse sein
    - **relaxed**
      DKIM; Es darf mit einer übergeordneten Domäne signiert werden.
      SPF: Der Return-Path und Header-From müssen in derselben „Organisationsdomäne“ sein.

DMARC-Alignment

- DMAR
  - Bei D
  - DMA
    - st
      D
      S
    - re
      D
      S

Nein, RFC 7489, Section 3.1 wird auch nach dem dritten Lesen nicht besser:

In relaxed mode, the [SPF]-authenticated domain and RFC5322.From domain must have the same Organizational Domain. In strict mode, only an exact DNS domain match is considered to produce Identifier Alignment.

Jede Mail, bei der der Return Path und der Header-From: in unterschiedlichen Organizational Domains sind, wird durch eine DMARC-Validierung fallen.

Oder auch “Warum Dienstleister S diesen komischen CNAME in unserer Domain haben will:

```
em1057.uv.uni-kiel.de.    IN    CNAME    u29988855.wl106.sendgrid.net.
em4072.uv.uni-kiel.de.    IN    CNAME    u29960932.wl183.sendgrid.net.
```

→ SPF-Alignment mit der Brechstange.

DMARC-Alignment (relaxed)

SPF

```
MAIL FROM:
  <sender@users.example.com>

From: sender@lists.example.com
```

DKIM

```
DKIM-Signature: v=1; ...;
d=example.com; ...

From: sender@child.example.com
```

Alignment und das Sender Rewriting Scheme

Reines SPF

- Prüft den Return Path
- Trick für Weiterleitungen: Umschreiben des Absenders
- Sender Rewriting Scheme (SRS)

user@woanders.local →

SRS0=HHH=TT=woanders.local=user@hier.example.orgOC

Alignment und das Sender Rewriting Scheme

SRS0=HHH=TT=woanders.local=user@hier.example.org

- Wiederherstellung der SPF-Validierung durch Umschreiben von Weiterleitungen
- SRS bricht das SPF-Alignment für DMARC
- Aber: Die DMARC-SPF-Validierung des Header-From: schlägt nach der Weiterleitung ohnehin fehl

  → Schadet also nicht

Richtlinie

```
"p=quarantine; "
"sp=reject; "
```

Was soll mit nicht-authentifizierten Nachrichten geschehen?

- none, quarantine oder reject
- sp=
  wird auf Subdomains angewendet ...
  … die aber auch eigene Records haben dürfen, die dann bevorzugt gelten

DMARC-Reporting

- rua
  - Aggregate Report
  - Regelmäßig (im Interval ri=…), auch für korrekt authentifizierte Mails
  - Mit komprimierten (gzip oder zip) XML-Anhang
  - Achtung: Es kommt ein Report pro eigener Domäne pro Absender.
    → Also ggf. so ein paar hunderte am Tag ...
  - Achtung: zwei unterschiedliche Schemata
    - Standard: Ohne Namespace oder http://dmarc.org/dmarc-xml/0.1
    - Draft: urn:ietf:params:xml:ns:dmarc-2.0

→ Auf keinen Fall daran denken, diese manuell auswerten zu wollen

DMARC-Reporting

- ruf
  - Forensic Report (RFC6591)
  - Sofort, wenn eine Nachricht aus der eigenen Domäne nicht korrekt authentifiziert werden konnte
  - Als Abuse Reporting Format (ARF, RFC 5965)
  - multipart/report mit entsprechenden Headern in einem message/feedback-report.

DMARC-Records

- Wann soll ein Fehler-Report erzeugt werden?
- Explizit nicht: Wann trat tatsächlicher ein Authentifizierungsfehler auf

fo=0	Report nur, wenn sowohl SPF als auch DKIM fehlschlagen
fo=1	Report, wenn entweder SPF oder DKIM fehlschlagen
fo=d	Report, wenn DKIM-Signatur inkorrekt ist. Keine Alignment-Prüfung!
fo=s	Report, wenn SPF-Validierung fehlschlägt. Keine Alignment-Prüfung!

Mailinglisten

- Mailinglisten sind jetzt „noch“ kaputter
  - SPF-Validierung des Header-From: schlägt bei klassischen Listen systembedingt fehl
  - SPF-Alignment ist nur bei Listenservern mit derselben Org-Domäne wie der Absender überhaupt möglich
  - DKIM-Validierung überlebt, wenn man signierte Header (Betreff) nicht anfasst
- Workarounds
  - Sympa:
    From: „Name via liste“ <liste@...>)
    From: „Name (e-mail)“ <liste@...>)
  - ARC (aber ….)
  - Siehe Diskussion vom 04.03.2025ff auf dem mail-ak-Verteiler

Christian-Albrechts-Universität zu Kiel

Spoofing? - Verfahren zur Absenderprüfung bei E-Mail

- **Spoofing und Absenderprüfung**
- **Wir versenden Mails nur von … (SPF)**
- **… und signiert sind unsere Mails auch (DKIM)**
- **… und man darf sich darauf verlassen (DMARC)**
- **… und da war noch das mit den Maillinglisten (ARC)**

Authenticated Receive Chain (ARC)

- Idee
  - Zwischenstationen dokumentierten (kryptographisch) ihre Validierungsergebnisse
  - Validierungsergebnis wird in speziellen Headern vermerkt
    - ARC-Authentication-Results: Von Signierenden bestimmter Authentifizierungsstatus (DKIM/SPF der Nachricht)
    - ARC-Message-Signature (Signatur der ggf. veränderten Nachricht)
    - ARC-Seal: Signatur über die ARC-Header
  - Siegelkette ist rückwärts überprüfbar
  - Übernimmt vieles von DKIM
- Benötigt eine Freigabeliste vertrauenswürdiger Versender
- Persönliche Einschätzung:
  Vermutlich nicht der Weisheit letzter Schluss

Danke

Vielleicht war ja was Neues dabei ...