Authentifizierung für E-Mail Sender

Florian Vierke – florian.vierke@mapp.com Sebastiaan de Vos – sebastiaan@inboxsys.com Michael Kliewe – m.kliewe@team.mail.de
– Version 0.8, 15.03.2023

Inhaltsverzeichnis

Dokumentengeschichte

Titel	**Titel**	**Datum**	**Verfasst durch**
Kürzel	Authentifizierung für E-Mail Sender	15.03.2023	Florian Vierke, Sebastiaan de Vos, Michael Kliewe

Version	**Datum**	**Beschreibung**	**Kürzel**
0.7	02.12.2022	Update	SdV, MK
0.6	23.11.2022	Kleine Änderungen	FV
0.5	11.11.2022	Strukturänderung	SdV
0.4	07.11.2022	Auswerten von DMARC-Reports	SdV
0.3	28.09.2022	Ergänzungen zu DMARC, Sections 4.1 und 4.2	MK
0.2	28.09.2022	Section 3 und 4 als Draft hinzugefügt	FV
0.1	20.07.2022	Erstes Draft	FV

Das Fälschen von Absenderadressen und damit das Vorspiegeln einer falschen Identität ist eine der häufigsten Formen des Betrugs in E-Mails. Indem Angreifer sich als jemand anderes ausgeben, wollen sie ihrem Opfer Informationen entlocken (z. B. Phishing) oder dieses dazu bewegen, eine für die Angreifer nützliche Handlung (z. B. CEO-Fraud) zu begehen. Dies führt auf Seiten der Opfer zu Misstrauen in E-Mail im Allgemeinen und es verursacht großen wirtschaftlichen Schaden für Privatpersonen wie auch für Unternehmen. In den vergangenen Jahren haben E-Mail-Experten deshalb mehrere Methoden entwickelt, um diese Form des Missbrauchs einzudämmen.

Dieses Dokument betrachtet E-Mail Authentication aus Sicht eines versendenden Mailsystems. Es nennt Konfigurationsbeispiele für SPF, DKIM und DMARC, damit E-Mail vor der Annahme authentifiziert, Identitätsmissbrauch erkannt und die EmpfängerInnen vor missbräuchlichen Nachrichten geschützt werden können. Zudem behandeln wir Softwarebeispiele, die dabei helfen, DMARC-Reports auszuwerten. Ziel ist, nur Nachrichten zu versenden, die den senderseitigen Richtlinien für SPF, DKIM und DMARC gerecht werden.

E-Mail Authentication für Empfänger?
Es ist ebenso wichtig, die Herkunft/Authentifizierung eingehender E-Mails zu prüfen und für eingehenden E-Mails DMARC-Reports zu versenden.

Was dazu getan werden muss, wird aber nicht in diesem Dokument behandelt. Informationen hierzu finden sich unter https://www.eco.de/themen/e-mail/downloads/email-authentication/. Dieses Dokument richtet sich besonders an Empfänger.

Die nachfolgenden Abschnitte zeigen, wie Sie SPF, DKIM und DMARC nutzen, um die E-Mails, die Sie versenden, zu authentifizieren.

Terminologien

Brief	**E-Mail Part**	**Bezeichnung laut RFC**	**Bezeichnung in dieses Dokument**
Absender am Briefumschlag	Message Envelope	RFC5321.MailFrom	Envelope Sender
Empfänger am Briefumschlag	Message Envelope	RFC5321.RcptTo	Empfänger
Absender auf Brief	Message Header	RFC5322.From	From-Header

1. Warum authentifizieren?

Eine Grundvoraussetzung, um überhaupt sinnvoll kommunizieren zu können (sei es per E-Mail oder über einen anderen Kanal) ist es, dass die beiden Kommunikationspartner bekannt und vertrauenswürdig sind. Im Fall von E-Mail ist es von entscheidender Bedeutung, ob wir den Absender kennen – und prüfen können, dass es sich tatsächlich um den Kommunikationspartner handelt, der er vorgibt zu sein. Ohne erfolgreiche Prüfung ist der Inhalt der Nachricht wertlos, gegebenenfalls sogar gefährlich.

Für Mailboxprovider ist Spambekämpfung eine zentrale Herausforderung. Daher gehen immer mehr Mailboxprovider auf der Empfängerseite dazu über, nur noch (Massen-)E-Mails von authentifizierten Versanddomains anzunehmen.

1.1. Warum mit DMARC authentifizieren?

Um Missbrauch der eigenen Versanddomain vorzubeugen, ist es wichtig, die Grundidee von DMARC zu verstehen: DMARC veröffentlicht Richtlinien für den Umgang von Verstößen gegen SPF und DKIM. Hierbei erfordert DMARC, dass eine E-Mail mit mindestens einer der beiden Methoden SPF oder DMARC konform ist. Falls beide Methoden fehlschlagen, gilt eine E-Mail als nicht authentisch.

Wenn ein Angreifer eine fremde Domain für einen illegitimen Nachrichtenversand missbraucht, kann die Zustellbarkeit darunter leiden. DMARC dient dazu, Missbrauch zu verhindern.

An dieser Stelle setzt die Policy an, die DMARC mit Hilfe des `p`-tags im DMARC Eintrag im DNS der `From:`-Header Domain veröffentlicht. Drei Werte sind für das `p`-tag zulässig:

none

Ist `none` gesetzt, fordert die im `From:`-Header angegebene Senderdomain, dass nichts unternommen werden soll, wenn es zu Verstößen gegen SPF und DKIM kommt.

quarantine

Ist `quarantine` gesetzt, fordert die im `From:`-Header angegebene Senderdomain, dass die Nachricht zwar angenommen, aber nicht direkt in den Posteingang zugestellt, sondern in Quarantäne, z. B. den SPAM-Ordner, gelegt werden soll.

reject

Ist `reject` gesetzt, fordert die im `From:`-Header angegebene Senderdomain, dass die Annahme der Nachricht verweigert und diese nicht zugestellt werden soll.

Nur mit einer "reject"-Policy ist DMARC vollständig aktiviert. DMARC mit "reject"-Policy erlaubt es Empfängern, neben IP-Reputation auch die Domain-Reputation in Betracht zu ziehen. So wird die Zustellbarkeit unter Umständen weniger aufgrund der - oft geteilten - IP-Reputation in Mitleidenschaft gezogen.

BIMI
Brand Indicators for Message Identification (BIMI) ermöglicht es, an einer vordefinierten Stelle im DNS ein Markenlogo sowie ein Zertifikat zu verlinken. Mailboxprovider auf Empfängerseite können dieses Logo in E-Mail-Programmen oder Webmail-Oberflächen einbinden und als Absenderlogo anzeigen.

BIMI setzt DMARC mit "reject"-Policy auf der Organisational Domain voraus.

---

2. Implementierung

Die drei wichtigsten Methoden werden kombiniert eingesetzt, um a) für eine Envelope-Sender-Domain die sendenden Systeme zu legitimieren (SPF), b) die Identität einer Domain zu verifizieren (DKIM) und c) um eine Richtlinie (DMARC) festzulegen, wie mit Nachrichten verfahren werden soll, welche SPF und DKIM nicht gerecht werden, sowie um Reports über den aktuellen Status möglichen Identitätsmissbrauchs zu erhalten. Die drei genannten Methoden werden unter dem Begriff "Email Authentication" zusammengefasst.

Email Authentication
Das Medium E-Mail enthält in seiner ursprünglichen Form keine Möglichkeit der Authentifizierung. Wir sehen nur die IP-Adresse des letzten, weiterleitenden Servers. Dies muss jedoch nicht zwangsläufig auch der versendende Mailserver sein. Für Domains gibt es nativ keinerlei Kontrollmöglichkeiten.

"Email Authentication" kombiniert die Methoden SPF, DKIM und DMARC zu einem Mechanismus, mit

> dem eingehende E-Mails auf ihre Authentizität geprüft werden können. Die Methoden stellen für sich genommen die folgenden Möglichkeiten zur Verfügung:

2.1. SPF

Mittels SPF wird in einem DNS TXT Eintrag hinterlegt, welche IPs für die jeweilige Versanddomain versenden dürfen. Das Setzen eines SPF DNS Eintrags ist einfach und sollte von Versendern implementiert werden.

Benötigt werden alle IP-Adressen bzw. IP Ranges, die zum Versand genutzt werden. Diese werden durch Leerzeichen getrennt in einen SPF Eintrag übernommen. Der fertige Eintrag könnte in etwa so aussehen:

```
DOMAIN.TLD TXT "v=spf1 ip4=192.0.2.0 ip4=192.1.2.0/24 ip6=fe80::0202:b3ff:fe1e:8329/64 include:sub.example.com -all"
```

Es gibt auch die Möglichkeit ein Redirect zu machen:

```
DOMAIN.TLD TXT "v=spf1 redirect:sub.example.com"
```

Wenn man ein Redirect setzt, sind - abgesehen von "v" - keine weitere Parameter erlaubt. Z.B. -all oder ~all sollte in dem Fall auch weggelassen werden! Letzteres ist ein häufig gemachter Fehler.

> *SPF Version*
> Für SPF existiert nur eine Version: spf1. Die Version "spf2.0/*" ist erstens keine SPF Version, sondern SenderID und zweitens obsolet. Mehr Information dazu: http://www.open-spf.org/SPF_vs_Sender_ID.

Oben genannte Beispiele sind eben nur Beispiele. SPF ist komplizierter als das! Für weitere Details zur Implementierung verweisen wir auf https://www.rfc-editor.org/rfc/rfc7208.

Leider schlägt SPF in vielen zentralen Anwendungsfällen von E-Mail fehl, beispielsweise bei Weiterleitungen oder der Nutzung von Mailinglisten. Daher wird SPF fast ausschließlich in Kombination mit anderen Authentifizierungsmethoden verwendet.

2.2. DKIM

Viel wichtiger ist somit die Implementierung von **DKIM**. Bei dieser Authentifizierungsmethode wird die E-Mail mit einem privaten Schlüssel signiert und der dazu passende öffentliche Schlüssel im DNS hinterlegt. Empfänger können nun prüfen, ob die Signatur der eingehenden E-Mail gültig ist. Ist das der Fall, ist sichergestellt, dass:

1. Der Versender den privaten Schlüssel besitzt
2. Der Versender Zugriff auf das DNS der signierenden Domain besitzt
3. Der Inhalt der empfangenen E-Mail (zumindest der signierten Teile) auf dem Transportweg von keinem weiterleitenden Mailserver verändert wurde

Wird *aligned* signiert, sprich die Domain im `From:`-Header der E-Mail und die DKIM-Domain gehören zu derselben Organisational Domain, können wir somit sicherstellen, dass der Absender der E-Mail auch tatsächlich vom Domainbesitzer zum Versenden berechtigt wurde.

So kann nicht nur die Integrität, sondern auch die Authentizität festgestellt werden.

Der öffentliche Teil des Schlüssels (Public Key) wird im DNS der signierenden Domain hinterlegt. Dazu ist ein Selektor notwendig. So sieht ein DKIM Key im DNS aus:

```
SELECTOR._domainkey.DOMAIN.TLD TXT "v=DKIM1; k=rsa; p=PUBLIC_KEY"
```

Somit ist es möglich, in einer Domain mehrere DKIM Keys - mittels unterschiedlichen Selektoren - zu setzen. Empfohlen wird auch, Keys und Selektoren regelmäßig zu rotieren und - für größere Organisationen - unterschiedliche Key/Selektor Paare für unterschiedliche Einsatzzwecke zu verwenden.

Es wird empfohlen, mit der Versanddomain im FROM:-Header ebenfalls DKIM zu signieren (*aligned*), da dies ein Erfordernis von DMARC ist.

Abgesehen von v- (Version) k- (Verschlüsselungstyp) und p-Parameter (Key), gibt es noch weitere Parameter die man setzen kann. Derzeit gibt es nur die Version "DKIM1". Neben RSA kann man auch ED25519 für die Verschlüsselung verwenden. Für weitere Details zur Implementierung verweisen wir auf https://www.dkim.org.

Die Implementierung von DKIM Signaturen kann mit verschiedenen Lösungen erfolgen, beispielsweise OpenDKIM oder rspamd.

2.3. DMARC

DMARC erfordert für eine Nachricht eine erfolgreiche Authentifizierung per SPF oder DKIM der From:-Header Domain. Darüber hinaus legt DMARC fest, welche Richtlinie bei Verletzungen von SPF und DKIM angewandt werden soll **und** ermöglicht durch Hinterlegung einer Kontaktadresse den Empfang von sog. Feedback Reports über die Authentifizierungsergebnisse einer Domain.

Hierfür muss ein DNS TXT Record unter _dmarc.<domain> zu finden sein, der in etwa wie folgt aussieht:

```
_dmarc.DOMAIN.TLD TXT "v=DMARC1; p=reject; rua=mailto:<reporting-address>"
```

An <reporting-address> gehen nun von teilnehmenden Mailboxprovidern i.d.R. tägliche DMARC-Reports im XML Format ein. Es wird empfohlen, diese grafisch auszuwerten. Wie das geht wird weiter unten beschrieben.

Failure Reports
Neben "rua" ist auch ein "ruf" Parameter erlaubt. Auch hier wird eine Reporting Adresse angegeben. An diese Reporting Adresse werden ad-hoc Fehler Reports gesendet. Die Verwendung des RUF-Parameters ist in Europa aus Datenschutzgründen umstritten, da in Failure Reports die gesamte E-Mail inklusive Betreff und Inhalt enthalten ist.

Häufiger Fehler bei Mailing-Listen, die zu DMARC-Problemen führen:

- Das Beibehalten des From:-Headers,
- Hinzufügen des Listen-Namens "[Listenname]" im Betreff
- Hinzufügen von Footer-Text im Body-Text,
- oder das Hinzufügen eines Reply-To:-Headers, der durch Oversigning gesichert "nicht-existent" war.

Bevor man p=reject verwenden kann, sollte man das setup zuerst testen. Dazu fangt man an mit ein Record der de-facto wirkungslos ist:

```
_dmarc.DOMAIN.TLD TXT "v=DMARC1; p=none; rua=mailto:<reporting-address>"
```

Mit p=none bekommt man zwar Reports, aber die Domain ist nicht mit DMARC geschützt.

Zunächst sollten die eingehenden DMARC-Reports über einen Zeitraum von 2-3 Wochen überprüft werden. Sobald man als

Versender sicher ist, ausschließlich gültig authentifiziert zu versenden, wird empfohlen, die Policy p=reject einzusetzen, um missbräuchlichen Versand Fremder über die eigene Domain zu unterbinden.

Oben genannte Beispiele sind eben nur Beispiele. Abgesehen von die genannte Parameter sind viele weitere Parameter möglich. Für weitere Details zur Implementierung verweisen wir auf https://www.rfc-editor.org/rfc/rfc7489

2.4. Empfangen von DMARC-Reports

Unterscheiden sich Reporting-Domain (RUA/RUF) und Versand-Domain, MUSS die RUA-Domain mittels einer zusätzlichen Subdomain (TXT _report._dmarc.senderdomain.com) dazu authentifiziert werden, Reports für die betreffende Domain zu empfangen. Verifying External Destinations in RFC 7489 geht ausführlich auf diese Notwendigkeit ein.

Der DMARC-DNS-Record kann mit Hilfe von Online-Check-Tools geprüft werden, beispielsweise:

- mimecast DMARC Record Check
- dmarcian DMARC Record Checker
- MxToolbox DMARC Check Tool
- InboxSys Domainchecker

Das Postfach, das DMARC-Reports empfängt, sollte:

- groß genug sein für die erwartete Menge E-Mails.
- das Empfangs-Ratelimit hoch genug haben, um - selbst gegen Mitternacht - auch größere Mengen E-Mails pro Minute zu empfangen
- einige Spam-Prüfungen deaktiviert haben, denn DMARC-Reports enthalten .xml Anhänge mit IP-Adressen, die auf Blacklists stehen könnten.
- Anhänge vom Typ .gz oder .xml erlauben.

Man sollte regelmäßig prüfen, ob das Postfach existiert/aktiv ist und E-Mails empfängt, denn für Versender von DMARC-Reports ist es recht lästig, wenn eine Ziel-Adresse nicht erreichbar ist und Bounces produziert werden, weil z.B. das Postfach voll ist, keine .gz Anhänge angenommen werden, E-Mails wegen Spam-Klassifizierung abgelehnt werden, usw.

2.5. Auswerten von DMARC-Reports

Die empfangenen DMARC-Reports sollten auch ausgewertet werden. Diese Auswertung sollte automatisiert passieren. Dazu gibt es unterschiedliche kommerzielle Werkzeuge, z.B. Agari, DMARCIAN oder DMARCadvisor, aber auch einige kostenlose Open Source Programme, die dabei behilflich sein können. Einige dieser kostenlosen Möglichkeiten möchten wir hier näher beleuchten:

2.5.1. ParseDMARC

ParseDMARC ist ein kleines Python Modul, dass DMARC-Reports aus einem IMAP-Postfach in eine Elasticsearch Datenbank importieren kann. Danach kann man die DMARC Ergebnisse auf einem Kibana- oder Splunk-Dashboard einsehen. Unter https://domainaware.github.io/parsedmarc/ findet sich eine ausführliche Dokumentation. Es gibt auch einen fertigen Docker-Stack mit parsedmarc, Elasticsearch und Kibana zum schnellen Ausprobieren.

2.5.2. DMARC Viewer

DMARC Viewer basiert auf Django und Python und es importiert DMARC-Reports aus einem E-Mail-Postfach in eine Postgres Datenbank. Dieses Tool hat ein integriertes Webinterface. Unter https://github.com/dmarc-viewer/dmarc-viewer/ befindet sich die Dokumentation zu dieses Tool.

2.5.3. DMARC Report

DMARC Report enthält einerseits einen Parser in Python und andererseits einen Viewer in PHP. Es basiert auf John levine's rddmarc Script. Dokumentation und weiterführende Links befinden sich hier: https://www.techsneeze.com/dmarc-report/

2.6. Interpretation der Auswertung

In diesen DMARC Auswertungstools sieht man nicht nur die E-Mails, die man selbst versendet hat, sondern auch, was Dritte mit dieser Domain versendet haben. Das können Mailinglisten oder Weiterleitungen sein, aber auch mißbräuchliche Verwendung der Domain.

Es gibt verschiedene Kriterien nach denen man filtern kann. Z.B. kann man nach IPs filtern wo DMARC fehlgeschlagen ist. Das können eben eigene, oder fremde, IPs sein. Von hier aus kann man weiter filtern, z.B. nach alignment Fehlern, fehlende Signaturen und ggf. auch SPF fails.

Version 0.8
Last updated 2023-03-24 17:29:16 +0100