RFC 9989 : Domain-based Message Authentication, Reporting, and Conformance (DMARC)

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 20 mai 2026

Date de publication du RFC : Mai 2026

https://www.bortzmeyer.org/9989.html

DMARC est une technique d'authentification du courrier électronique qui permet à un domaine d'indiquer quelle est sa politique de sécurité vis-à-vis des messages dont l'expéditeur (le champ `From:` de l'en-tête) indique ce domaine. Il donne au gérant du domaine la possibilité d'annoncer sa politique de sécurité « tous les messages de ce domaine sont authentifiés (via SPF ou DKIM) ». Typiquement, DMARC est le couronnement d'une démarche de sécurité du courrier, ce qu'on annonce quand on a bien tout authentifié. Par contre, attention, en authentifiant la donnée visible par les utilisateurs et pas les données techniques, il casse certains usages du courrier. DMARC était à l'origine normalisé dans le RFC 7489 [1], que ce nouveau RFC remplace. Mais rassurez-vous si vous avez déjà déployé DMARC : les changements ne sont pas radicaux. Le principal est le nouvel algorithme pour trouver l'enregistrement DMARC pertinent (celui à l'apex du domaine enregistré).

Revenons sur les problèmes de sécurité du courrier électronique. Un message typique, tel que normalisé par le RFC 5322, comprend dans son en-tête ce genre d'informations :

```
Date: Wed, 18 Mar 2026 17:28:36 +0800
From: Jiankang Yao <yaojk@cnnic.cn>
To: 125attendees@ietf.org
Subject: [125attendees] Wednesday 8:00 pm, Shenzhen light show for IETF 125
X-Mailer: iPhone Mail (21D61)
[et bien d'autres]
```

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc7489.txt`

Une qui nous intéresse particulièrement est l'expéditeur. Cette notion est plus compliquée qu'il n'y parait (il y a plusieurs définitions possibles de « expéditeur ») mais pour DMARC, c'est simple : le champ qui nous intéresse est uniquement le `From:` (section 3.6.2 du RFC 5322). C'est en effet celui qui est typiquement affiché par les MUA, et c'est celui que DMARC va protéger. On l'appelle souvent RFC5322-From pour le distinguer de celui qui apparait dans l'enveloppe du courrier, le RFC5321-From (et qui n'est pas montré dans mon exemple).

Les techniques d'authentification existantes avant DMARC, SPF (RFC 7208) et DKIM (RFC 6376), n'authentifient **pas** ce champ mais d'autres (le RFC5321-From pour SPF et le domaine indiqué dans la signature pour DKIM), qui ne sont pas en général affichés à l'utilisateurice final·e. DMARC va permettre d'utiliser ces deux techniques, SPF et DKIM, pour les appliquer à l'expéditeur (RFC5322-From). Un test DMARC réussi signifie que SPF **ou** DKIM a réussi mais **aussi** que le domaine authentifié par SPF ou DKIM est le même que celui présent dans le `From:` ; on parle d'**alignement** du nom de domaine. Cela ne va pas de soi car il y a de nombreux usages **légitimes** du courrier où ces domaines ne sont pas alignés, et DMARC casse donc ces usages.

Dans les exemples de messages reçus après traitement par DMARC, on va regarder les champs `Authentication-Results`. Normalisés dans le RFC 8601, ils sont ajoutés par le récepteur et indiquent le résultat d'une technique d'authentification. Ici, un exemple où SPF et DKIM ont marché (tous les exemples ici sont réels, issus de mes boites aux lettres) :

```
Authentication-Results: mail.bortzmeyer.org; dmarc=pass (p=quarantine dis=none) header.from=afnic.fr
Authentication-Results: mail.bortzmeyer.org;
        dkim=pass (2048-bit key; secure) header.d=afnic.fr header.i=@afnic.fr header.a=rsa-sha256 header.s=a
        header.b=bdGM6W8o;
        dkim-atps=neutral
Authentication-Results: mail.bortzmeyer.org; spf=pass (sender SPF authorized) smtp.mailfrom=afnic.fr
        (client-ip=2001:67c:2218:10::51:1; helo=mx1.nic.fr; envelope-from=quelqu.un@afnic.fr; receiver=bortz
```

Le domaine `afnic.fr` a bien été authentifié, à la fois par SPF et par DKIM, et DMARC passe donc (le champ `From:` n'est pas montré ici mais il indiquait bien une adresse `@afnic.fr`).

Il est également important de se souvenir que DMARC ne fait qu'authentifier le domaine, il ne garantit pas que le message soit sincère, sûr, utile ou quoi que ce soit d'autre. Si on reçoit un message de Trump, on peut prouver qu'il vient bien de `whitehouse.gov` mais il sera quand même certainement mensonger. C'est pour cela qu'il est absurde, comme on le lit dans certains forums, de dire « je ne comprends pas, j'ai bien mis un enregistrement DMARC et mes messages finissent quand même dans la boite Spam » : les spammeurs font du DMARC, eux-aussi.

L'inverse est vrai aussi, un message légitime et désiré peut parfaitement échouer au test DMARC, d'autant plus, que, comme indiqué plus haut, DMARC casse plusieurs usages légitimes du courrier. Il vaut donc mieux ne pas refuser un message uniquement sur la base d'un échec DMARC mais traiter cet échec comme une indication parmi d'autres. Ce RFC 9989 insiste sur ce point (notamment sa section 7), en mentionnant également le RFC 7960, qui détaille les problèmes venant de l'utilisation de DMARC.

La section 2 du RFC détaille le cahier des charges de DMARC. Comme avec toutes les solutions de sécurité, il faut garder en tête ce cahier des charges lorsqu'on évalue DMARC. Aucune solution de sécurité n'est parfaite : elles collent simplement plus ou moins bien à leur cahier des charges. Celui-ci, en résumé, est :

— Permettre aux gérants de noms de domaine d'annoncer leur politique d'authentification du courrier et leurs souhaits quant au traitement du courrier qui ne passerait pas cette authentification.
— Fonctionner dans le contexte de l'Internet (donc, sans autorité centrale).
— Traiter uniquement les cas où le message malveillant copie exactement un nom de domaine qu'on gère. En d'autres termes, les usurpations utilisant des noms qui ressemblent (`goog1e.com` au lieu de `google.com`) sont hors-sujet.
— Authentifier uniquement le nom de domaine qui est dans l'adresse (RFC 5322, section 3.4). En d'autres termes, dans un `From:` « Emmanuel Macron `<emmanuel5561@gmail.com>`, DMARC ne se préoccupe que du `gmail.com`, pas du `emmanuel5561`.
— Authentifier uniquement l'adresse, pas le nom affiché (« Emmanuel Macron » dans l'exemple ci-dessus). La section 11.4 rappelle ce point très important.

Un bon cahier des charges a une autre section très importante : celle des non-objectifs, des choses qu'on n'essaie pas de faire. (Regardez les documents commerciaux : ils n'ont jamais l'honnêteté de lister ce qu'ils ne font pas.) Pour DMARC :

— Il ne dit évidemment rien sur les domaines qui ont choisi de ne pas avoir d'enregistrement DMARC dans le DNS. Dit autrement, DMARC est "*opt-in*".
— Il n'essaie pas de s'occuper des autres informations présentes dans l'en-tête (comme `Reply-To:` ou `Date:`).
— Il ne s'occupe pas des tricheries sur le nom affiché, comme dans l'exemple « Emmanuel Macron » plus haut (ou bien, tiré de ma boite Spam `From: "amendes.gouv.fr" <amendes-gouv-nepasrepondr`
Tant pis pour ceux et celles qui s'obstinent à utiliser un logiciel qui, par défaut, n'affiche que ce nom (Outlook fait encore ça). Relisez la section 11.4 du RFC.
— Et naturellement, DMARC ne s'occupe pas du contenu du message, qu'il soit mensonger (« je suis l'ex-ministre des finances du Nigéria ») ou malveillant (logiciel qui va tenter d'exploiter une faille de sécurité pour prendre le contrôle de votre ordinateur).

Par exemple, voici un spam, prétendant venir de l'ANTAI mais qui passe tous les tests (le nom de domaine dans l'adresse n'a rien à voir avec l'ANTAI mais beaucoup d'utilisateurs n'y feront pas attention, et ce nom avait bien un enregistrement DMARC) :

```
Authentication-Results: mail.bortzmeyer.org; dmarc=pass (p=quarantine dis=none) header.from=rtm.gov.my
Authentication-Results: mail.bortzmeyer.org;
        dkim=pass (2048-bit key; secure) header.d=rtm.gov.my header.i=@rtm.gov.my header.a=rsa-sha256 header.s=r
        header.b=MaKApt+s;
        dkim-atps=neutral
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=rtm.gov.my; s=rtm; t=1775231898; x=1775836698; darn=bortzmeyer.org;
        h=content-transfer-encoding:mime-version:subject:message-id:to:from
        :date:from:to:cc:subject:date:message-id:reply-to;
        bh=L6c3f6fDBrQyBjkO1RiiF3vHHNmYjLDd4A1urn+cNJg=;
        b=MaKApt+s0vgkVsgh3VoFE0/MgCt8ilWkghyQKUbj2NnhgInADkR8G3aW0UbklkuDDV
 ...
Authentication-Results: mail.bortzmeyer.org; spf=pass (sender SPF authorized) smtp.mailfrom=rtm.gov.my
        (client-ip=2607:f8b0:4864:20::f64; helo=mail-qv1-xf64.google.com; envelope-from=antai.gouv.fr@rtm.gov.my
        receiver=bortzmeyer.org)
Subject: ACTION REQUISE SOUS 24H
From: "Antai.gouv.fr" <Antai.gouv.fr@rtm.gov.my>
```

La section 3 du RFC décrit les termes utilisés par DMARC, entre autres :

— Domaine de l'auteur : ce qui est après l'arobase dans l'adresse indiquée par le champ `From:` de l'en-tête (rappel : pas celui de l'enveloppe).
— Domaine DKIM : celui indiqué par le paramètre `d=` dans la signature DKIM (rappel : le domaine DKIM peut n'avoir aucun rapport avec l'adresse de l'en-tête ou avec celle de l'enveloppe).
— Domaine SPF : ce qui est après l'arobase dans l'adresse indiquée par le champ `From` de l'enveloppe (rappel : pas celui de l'en-tête). Dans le contexte de DMARC, ce terme ne s'applique pas au domaine indiqué dans la commande EHLO (ou HELO) de SMTP.

— Titulaire du domaine : la personne physique ou morale qui décidé d'enregistrer un nom de domaine et qui le gère ensuite.
— Domaine organisationnel : le domaine au sommet du sous-arbre qui a la même administration (le RFC 5598 est une bonne lecture ici). Ainsi, `bortzmeyer.org` est le domaine organisationnel de `mail.bortzmeyer.org`. En pratique, c'est souvent le domaine enregistré du RFC 9499, domaine qui a été enregistré auprès d'un registre.
— Suffixe public : domaine où le public peut enregistrer un sous-domaine, par exemple `.fr` ou `eu.org`. Ainsi, dans `mail.foobar.eu.org`, `foobar.eu.org` est le domaine organisationnel et `eu.org` le suffixe public, ou domaine d'enregistrement.

Armé de cela (mais il y a d'autres termes, que je présenterai au fur et à mesure), on peut passer à la section 4, qui explique les concepts importants.

DMARC permet à un titulaire de domaine d'annoncer sa politique d'authentification d'un domaine de l'auteur d'un courrier. On n'authentifie donc que le domaine, pas toute l'adresse (je l'ai déjà dit mais c'est important). Et DMARC ne s'intéresse qu'à ce qu'il appelle le domaine de l'auteur, donc le `From:` dans l'en-tête (également appelé « RFC5322.From »). DMARC annonce juste une politique, l'authentification est faite avec SPF (RFC 7208) ou DKIM (RFC 6376).

Un concept essentiel dans DMARC est celui d'**alignement**. Il y a alignement quand le domaine authentifié par SPF (celui de l'enveloppe, le « RFC5321.From ») ou par DKIM (celui indiqué dans le `d=` de la signature) coïncide avec le domaine de l'auteur (celui du `From:` de l'en-tête). Plus précisément, il peut y avoir un alignement strict (les deux noms de domaine sont identiques) ou relâché (les deux noms sont dans le même domaine organisationnel). Le choix se fait dans l'enregistrement DMARC publié.

Justement, on le publie où ? Via le DNS, dans un enregistrement de type TXT, publié dans le sous-domaine `_dmarc`, par exemple :

```
% dig +short _dmarc.proton.me TXT
"v=DMARC1; p=quarantine; fo=1; aspf=s; adkim=s;"
```

On bénéficie ainsi de toute l'infrastructure, très fiable et éprouvée, du DNS. Ce sous-domaine `_dmarc` figure dans le registre IANA des noms commençant par un trait bas <https://www.iana.org/assignments/dns-parameters/dns-parameters.xml#underscored-globally-scoped-dns-node
créé par le RFC 8552.

Au passage, si vous voulez voir (ou enregistrer), sur un serveur DNS faisant autorité, uniquement les requêtes DNS pour le sous-domaine `_dmarc`, dnscap <https://www.dns-oarc.net/tools/dnscap> permet de le faire facilement :

```
% dnscap -g -x _dmarc
```

(Je triche un peu, le `-x` attrape en effet davantage que cela mais ça suffit en première approximation.)

Le format exact de l'enregistrement DMARC utilise des doublets clé=valeur, comme celui de DKIM. (Sa description formelle, en ABNF - RFC 5234 - est dans la section 4.8.) Les clés possibles figurent dans un registre IANA <https://www.iana.org/assignments/dmarc-parameters/dmarc-parameters.xml#tag>. Les plus courantes sont :

— `v` : c'est obligatoirement le premier doublet clé=valeur et il indique la version de DMARC, actuellement `DMARC1`.
— `p` : c'est la clé la plus importante, celle qui indique la politique à appliquer aux messages qui ne passent pas la validation DMARC. Une valeur `reject` indique que le titulaire du domaine recommande le rejet des messages invalides (cela ne peut être qu'une recommandation, car le récepteur du courrier reste évidemment libre de sa politique). `quarantine` recommande une mise en attente quelque part (un dossier « peut-etre-spam » par exemple). Enfin, `none` indique qu'on recommande de ne rien faire. Cela peut être utilisé quand on craint les conséquences de DMARC sur certains usages (cf. RFC 7960) mais qu'on pense que certains récepteurs vont mal traiter les messages des domaines sans DMARC. Ou bien cela peut être utile dans certains audits « de sécurité » qui demandent un enregistrement DMARC, n'importe lequel. Enfin, un `p=none` peut être utilisé lors d'un déploiement progressif de DMARC, quand on veut juste tester, avant de publier une politique plus fasciste. Comme DMARC permet de solliciter l'envoi de rapports d'erreur, un `p=none` peut être accompagné d'une telle sollicitation.
— `sp` : comme `p` mais pour les sous-domaines du domaine qui a l'enregistrement DMARC. Les valeurs possibles sont les mêmes que pour `p`.
— `np` : nouveauté, initialement décrite dans le RFC 9091. C'est le traitement à appliquer aux sous-domaines non-existants du domaine pour lequel une politique DMARC est publiée. Les valeurs possibles sont les mêmes que pour `p`.
— `ruf` : c'est ainsi qu'on sollicite l'envoi de rapports d'erreur. On indique les URI où envoyer ces rapports (souvent des URI de plan `mailto:`, pour demander des rapports par courrier). Le format des rapports est spécifié dans les RFC 9991, RFC 6651 et RFC 6552. `ruf` demande un rapport par message invalide, `rua` permet de demander des rapports agrégés. Leur format figure dans le RFC 9990.
— `psd` : nouveauté de notre RFC, s'il a la valeur `y`, il indique que le domaine est un suffixe public (PSD : "*Public Suffix Domain*"), c'est-à-dire un domaine dont les sous-domaines peuvent être délégués à d'autres entités (comme c'est le cas de `.re` ou `eu.org`).
— `fo` : diverses options pour la génération de rapports d'erreur.
— `adkim` et `adpf` : tous les deux peuvent prendre la valeur `s` (strict) ou `r` (relâché, ou laxiste). Ils indiquent si l'alignement avec l'identificateur authentifié par DKIM ou SPF doit être strict (les deux identificateurs sont rigoureusement identiques) ou relâché (l'identificateur authentifié peut se contenter d'être dans le même domaine enregistré que celui qui a un enregistrement DMARC). Par défaut, DMARC est laxiste. Notez que cela permet à quelqu'un qui peut utiliser un sous-domaine de se faire authentifier comme étant dans l'apex (section 11.8 du RFC). Demander un alignement strict résout ce problème (mais impose que vous controliez bien les sous-domaines).

Les clés inconnues doivent être ignorées, ce qui permet d'en ajouter de nouvelles sans tout casser. La politique pour un éventuel ajout est « Spécification nécessaire » (RFC 8126).

On a parlé du domaine organisationnel, l'apex du domaine testé par DMARC. C'est en fait une notion administrative, pas technique, ce qui fait que ce domaine organisationnel n'est pas évident à identifier dans le DNS. Pour le trouver, l'ancien RFC, le RFC 7489, suggérait de faire appel à une liste de suffixes publics, comme la PSL (rappel : il n'existe pas de liste officielle). Notre nouveau RFC suggère une autre méthode, en remontant l'arbre des noms de domaine. On commence par le domaine qu'on veut authentifier, et, si on n'y trouve pas de politique DMARC, on essaie son domaine parent et ainsi de suite, jusqu'à ce qu'on trouve un enregistrement DMARC. Ainsi, pour `truc.machin.example.com`, on essaiera successivement `_dmarc.truc.machin.example.com`, `_dmarc.machin.example.com`, `_dmarc.example.com` et enfin `_dmarc.com`. La dernière requête permet donc au registre de `.com` de définir une politique DMARC qui s'appliquera à tous les domaines sans politique DMARC. C'est très dangereux, pour les raisons expliquées dans le RFC 1535 mais cela permet de se passer de liste de suffixes publics, et c'est plus souple pour le cas des grosses organisations, qui peuvent avoir des politiques DMARC dans des sous-domaines qui ne sont pas délégués.

Notez que l'éventuelle présence de la clé `psd` va compliquer les choses mais je n'ai pas vraiment le courage de détailler ici l'algorithme complet.

Maintenant, voyons quels sont les acteurs d'un déploiement de DMARC. D'abord, le titulaire du domaine qui envoie des messages. Il doit publier un enregistrement SPF à l'apex du domaine. Il doit signer les courriers sortants avec DKIM (ce qui implique de publier les clés publiques DKIM dans le DNS). Notez que DMARC n'a pas besoin de SPF **et** de DKIM, un seul des deux suffit mais, bon, autant tout faire. Il a intérêt à créer une boite dédiée pour recevoir les rapports sur les messages invalides. Le titulaire doit enfin publier dans le DNS l'enregistrement DMARC (celui qui commence par `_dmarc`). Au début, on utilise typiquement une politique indulgente (`p=none`). Puis on teste.

Comment teste t-on ? En lisant les rapports indiquant des messages invalides (RFC 9990 et RFC 9991). Les rapports agrégés sont du XML, assez lisibles par un humain mais, en pratique, on préférera typiquement utiliser un programme qui les synthétisera dans une forme plus lisible. (Je n'utilise pas actuellement un tel programme, car je n'en ai pas trouvé. Il faut qu'il tourne en local - pas question de confier les rapports à un tiers - et ne nécessite pas d'installer toute une batterie de cuisine PHP et MariaDB.) Une fois qu'on a trouvé les problèmes (une application oubliée dans un coin qui envoie des courriers sans passer par les serveurs centraux…) et qu'on les a corrigés, on peut durcir la politique (`p=quarantine`, par exemple). Il est raisonnable d'attendre plusieurs semaines, voire mois, pour être sûr d'avoir vu tous les problèmes.

(Et si vous êtes gérant d'un suffixe public - - un domaine sous lequel d'autres entités peuvent enregistrer des noms, demandez-vous si vous devez publier du DMARC avec `psd=y`. Ce n'est pas obligatoire, cf. section 5.2.)

Et le récepteur du courrier, que doit-il faire ? Il extrait du message le domaine de l'auteur. Il cherche s'il y a un enregistrement DMARC. Il exécute les tests SPF et DKIM. S'il récupère un ou plusieurs domaines authentifiés, il vérifie l'alignement (strict ou relâché). Si au moins un domaine authentifié est aligné avec le domaine de l'auteur, le test DMARC est un succès. Sinon, c'est un échec. Notez bien qu'il n'est pas nécessaire que SPF et DKIM réussissent tous les deux. Si le test DMARC se termine en échec, on applique un traitement, qui dépend de la politique suggérée dans l'enregistrement DMARC, et de la politique propre du receveur. Voici un exemple où DMARC dit que tout s'est bien passé :

```
From: "Projet Arcadie" <admin@projetarcadie.com>
Authentication-Results: mail.bortzmeyer.org; dmarc=pass (p=none dis=none) header.from=projetarcadie.com
Authentication-Results: mail.bortzmeyer.org;
        dkim=pass (2048-bit key; unprotected) header.d=projetarcadie.com header.i=@projetarcadie.com header.
        header.b=plrnZlsJ;
        dkim=pass (2048-bit key) header.d=projetarcadie.com header.i=@projetarcadie.com header.a=rsa-sha256
        header.b=m84VgM5U;
        dkim-atps=neutral
Authentication-Results: mail.bortzmeyer.org; spf=pass (sender SPF authorized) smtp.mailfrom=projetarcadie.co
        helo=arcadieweb01.octopuce.fr; envelope-from=admin@projetarcadie.com; receiver=bortzmeyer.org)
```

Ici, il y avait un enregistrement SPF (qui autorise l'émetteur SMTP donc cela suffit), deux signatures DKIM, toutes les deux corrects, il y a alignement strict, et donc DMARC passe, il n'y a aucun doute que le domaine émetteur était bien `projetarcadie.com`. (La politique DMARC était `p=none` donc un éventuel échec n'aurait sans doute pas eu beaucoup de conséquences.)

Ici, DKIM a échoué (pourquoi ? mystère mais c'est peut-être la faute de SpamAssassin qui a modifié le sujet, il faudrait que je vérifie ma configuration) mais SPF réussit (le message vient bien de Gmail) donc DMARC est content (il s'agissait bien d'un spam, tentative d'escroquerie financière) :