

RFC 9228 : Delivered-To Email Header Field

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 14 avril 2022

Date de publication du RFC : Avril 2022

<https://www.bortzmeyer.org/9228.html>

Le courrier électronique est souvent plus compliqué et plus varié que ne le croient certains utilisateurs. Par exemple, rien ne garantit que l'adresse à laquelle un message est livré soit listée dans les champs de l'en-tête (RFC 5322¹) comme `To :` ou `Cc :`. Cette adresse de destination est en effet séparément gérée, dans le protocole SMTP (RFC 5321) et ses commandes `RCPT TO`. Et puis un message peut être relayé, il peut y avoir plusieurs livraisons successives. Bref, quand on a un message dans sa boîte aux lettres, on ne sait pas forcément quelle adresse avait servi. D'où ce RFC expérimental qui propose d'élargir le rôle du champ `Delivered-To :` pour en faire un bon outil d'information.

Ce champ existe déjà et on le voit parfois apparaître dans les en-têtes, par exemple dans un message que je viens de recevoir :

```
Delivered-To: monpseudo@sources.org
```

Mais il n'est pas vraiment normalisé, son contenu exact peut varier et, en pratique, lorsqu'il y a eu plusieurs livraisons successives suivies de transmissions à une autre adresse, il ne reste parfois qu'une occurrence de ce champ, la plus récente. Difficile donc de compter dessus.

Mais c'est quoi, la livraison d'un message? Comme définie par le RFC 5598, dans sa section 4.3.3, c'est un transfert de responsabilité d'un MTA vers un MDA comme procmail. Il peut y avoir plusieurs livraisons successives par exemple si le MDA, au lieu de simplement écrire dans la boîte aux lettres de

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc5322.txt>

l'utilisateur, fait suivre le message à une autre adresse. Un exemple typique est celui d'une liste de diffusion, où le message va être délivré à la liste puis, après transmission par le logiciel de gestion de listes, à chaque utilisateur (dont les MDA renverront peut-être le message, créant de nouvelles livraisons).

Pourquoi est-il utile d'avoir de l'information sur ces livraisons successives? Outre la curiosité, une motivation importante est le débogage. S'il y a des problèmes de livraison, la traçabilité est essentielle à l'analyse. Une autre motivation est la détection automatique de boucles, un problème souvent compliqué.

On l'a dit, `Delivered-To:` existe déjà, même s'il n'était normalisé dans aucun RFC, bien qu'il soit cité dans des exemples, comme dans le RFC 5193. Ce RFC 9228 enregistre le champ `<https://www.iana.org/assignments/message-headers/message-headers.xml#perm-headers>` (selon la procédure du RFC 3864), et prévoit d'étendre son application. (Une tentative de normalisation est en cours dans `draft-duklev-deliveredto`.) Du fait de cette absence de normalisation, il existe de nombreuses variations dans l'utilisation de `Delivered-To:`. Par exemple, si la plupart des logiciels mettent comme valeur une adresse de courrier et rien d'autre, certains ajoutent des commentaires. Notons que le champ `Received:` est souvent utilisé de la même façon, via sa clause `for`. On peut ainsi trouver (regardez le `for`):

```
Received: from bendel.debian.org (bendel.debian.org [82.195.75.100])
        (using TLSv1.3 with cipher TLS_AES_256_GCM_SHA384 (256/256 bits)
        key-exchange X25519 server-signature RSA-PSS (2048 bits) server-digest SHA256)
        (No client certificate requested)
        by ayla.bortzmeyer.org (Postfix) with ESMTPS id 6D245A00D9
        for <monpseudo@sources.org>; Wed, 16 Feb 2022 13:22:03 +0100 (CET)
```

Là encore, il n'y a pas de normalisation et il semble qu'on ne puisse pas compter sur cette clause `for`. Par exemple, elle n'indique pas forcément une livraison mais peut être utilisée pour une simple transmission d'un serveur à un autre.

Venons-en à la définition de l'utilisation de `Delivered-To:` proposée par notre RFC dans sa section 4:

- Le champ `Delivered-To:` est ajouté uniquement lors d'une livraison,
- sa valeur est l'adresse de courrier utilisée pour cette livraison (une adresse unique, jamais de liste),
- si une réécriture de l'adresse est faite, on rajoute un `Delivered-To:` avec l'adresse réécrite,
- comme il peut y avoir plusieurs `Delivered-To:` (en cas de livraisons multiples), le logiciel qui en ajoute un doit le mettre en haut (comme pour les autres champs de trace de l'en-tête, par exemple `Received:`, cf. RFC 5321, section, 4.1.1.4), l'ordre chronologique des `Delivered-To:` sera donc de bas en haut,
- et donc les logiciels ne doivent pas réordonner les `Delivered-To:` ou en supprimer, sous peine de casser cette chronologie.

En conséquence, si un logiciel qui va ajouter un `Delivered-To:` avec une certaine adresse voit un `Delivered-To:` avec la même adresse, il peut être sûr qu'il y a une boucle. Voici un exemple de deux `Delivered-To:` différents, en raison du passage par une liste de diffusion (le message comportait évidemment bien d'autres champs dans son en-tête). La liste a été la première à traiter le message, la livraison finale ayant eu lieu après (donc plus haut dans le message, cf. section 5 du RFC) :

```
Delivered-To: monpseudo@sources.org
Delivered-To: lists-debian-user-french@bendel.debian.org
```

Le RFC donne un exemple imaginaire mais plus détaillé (j'ai raccourci les exemples du RFC, référez-vous à lui pour avoir les `Received:` et autres détails). D'abord, le message est émis par Ann :

```
From: Ann Author <aauthor@com.example>
Date: Mon, 25 Jan 2021 18:29:00 -0500
To: list@org.example
Subject: Sending through a list and alias
Sender: Ann Author <aauthor@com.example>
```

Ce message a été envoyé à une liste de diffusion. Il est donc désormais :

```
Delivered-To: list@org.example
From: Ann Author <aauthor@com.example>
Date: Mon, 25 Jan 2021 18:29:06 -0500
To: list@org.example
Subject: Sending through a list and alias
```

Un des abonnés à la liste est `alumn@edu.example`. Chez lui, le message sera :

```
Delivered-To: alumn@edu.example
Delivered-To: list@org.example
From: Ann Author <aauthor@com.example>
Date: Mon, 25 Jan 2021 18:29:06 -0500
To: list@org.example
Subject: Sending through a list and alias
```

Et cet abonné fait suivre automatiquement son courrier à `theRecipient@example.net`. Le message, dans son état final, sera :

```
Delivered-To: theRecipient@example.net
Delivered-To: alumn@edu.example
Delivered-To: list@org.example
From: Ann Author <aauthor@com.example>
Date: Mon, 25 Jan 2021 18:29:06 -0500
To: list@org.example
Subject: Sending through a list and alias
```

Voilà, vous savez tout désormais sur l'extension proposée de l'utilisation de `Delivered-To:`. La section 6 attire toutefois notre attention sur quelques risques. `Delivered-To:` a comme valeur une donnée personnelle. Donc, attention, cela peut poser des questions de vie privée. Par exemple, la liste des adresses par lesquelles est passé un message peut en révéler plus que ce que le destinataire connaissait. Le problème risque surtout de se poser si quelqu'un fait suivre manuellement un message en incluant tous les en-têtes.

Autre problème potentiel, certains systèmes stockent les messages identiques en un seul exemplaire. Si on écrit à `alice@example.com` et `bob@example.com`, le serveur d'`example.com` pourrait décider de ne stocker qu'un seul exemplaire du message, avec des liens depuis les boîtes aux lettres d'Alice et Bob. Ce serait évidemment incompatible avec `Delivered-To:` (et pas question de mettre deux `Delivered-To:`, on ne veut pas révéler à Alice que Bob a reçu le message et réciproquement).