RFC 4871 : DomainKeys Identified Mail (DKIM) Signatures

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 24 mai 2007

Date de publication du RFC : Mai 2007

https://www.bortzmeyer.org/4871.html

Le courrier électronique, on le sait, n'offre, en natif, aucun mécanisme d'authentification des utilisateurs. Il est donc assez facile d'envoyer un courrier prétendant venir de Bill.Gates@microsoft.com. DKIM, spécifié à l'origine dans notre RFC, est la dernière tentative de combler ce manque. Ce RFC a ensuite été remplacé par le RFC 6376 [1].

Notons d'abord que, si le courrier électronique, tel que décrit dans les RFC 5321 ou RFC 5322, ou bien leurs prédécesseurs, n'offre pas d'authentification des utilisateurs, ce n'est pas parce que leurs concepteurs étaient imprévoyants ou trop confiants dans la nature humaine. C'est tout simplement parce que le problème est très complexe, ne serait-ce que parce qu'il n'existe pas de fournisseur d'identité unique sur Internet, qui pourrait jouer le rôle que joue l'État avec les papiers d'identité.

Mais l'absence d'authentification devient de plus en plus regrettable, d'autant plus qu'elle est activement exploitée par les spammeurs et les phisheurs pour leurs entreprises malhonnêtes. Non seulement il est difficile de retrouver le véritable envoyeur d'un message, mais une personne tout à fait innocente peut voir son identité usurpée. Ces problèmes de sécurité sont documentés dans le RFC 4686.

De nombreuses tentatives ont eu lieu pour tenter de traiter ce problème. Certaines ont connu un déploiement non négligeable comme PGP (normalisé dans le RFC 4880), surtout connu pour ses services de chiffrement mais qui peut aussi servir à l'authentification.

PGP est surtout utilisé dans des environnements à forte composante technique, au sein de groupes d'experts qui se connaissent. D'autres protocoles ont tenté de traiter le problème de l'authentification de la masse d'utilisateurs de Hotmail ou de Wanadoo.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6376.txt

SPF (normalisé dans le RFC 4408) a été condamné par l'hostilité de la plupart des gros acteurs.

DKIM, successeur de DomainKeys de Yahoo, après sa fusion avec l'IIM de Cisco, vient d'atteindre le statut de norme à l'IETF après de nombreux rebondissements. Comme SPF, il vise surtout à authentifier le **domaine** dans l'adresse électronique, en d'autres termes à garantir que le courrier vient bien de `microsoft.com`, laissant à Microsoft le soin de dire si la partie gauche (`Bill.Gates`) est authentique ou pas.

Mais, contrairement à SPF, il ne procède pas par énumération des adresses IP des MTA autorisés à émettre du courrier pour le compte d'un domaine mais par **signature** cryptographique.

Le principe de DKIM est le suivant (les exemples sont tirés de l'excellente annexe A du RFC). Un message émis ressemble à :

```
From: Joe SixPack <joe@football.example.com>
To: Suzie Q <suzie@shopping.example.net>
Subject: Is dinner ready?
Date: Fri, 11 Jul 2003 21:00:37 -0700 (PDT)
Message-ID: <20030712040037.46341.5F8J@football.example.com>

Hi.

We lost the game. Are you hungry yet?

Joe.
```

DKIM (qui peut être installé dans le MUA mais sera en général plutôt dans le MSA) le signe et ajoute un en-tête **DKIM-Signature** :

```
DKIM-Signature: v=1; a=rsa-sha256; s=brisbane; d=example.com;
      c=simple/simple; q=dns/txt; i=joe@football.example.com;
      h=Received : From : To : Subject : Date : Message-ID;
      bh=2jUSOH9NhtVGCQWNr9BrIAPreKQjO6Sn7XIkfJVOzv8=;
      b=AuUoFEfDxTDkHlLXSZEpZj79LICEps6eda7W3deTVFOk4yAUoqOB
        4nujc7YopdG5dWLSdNg6xNAZpOPr+kHxt1IrE+NahM6L/LbvaHut
        KVdkLLkpVaVVQPzeRDI009SO2Il5Lu7rDNH6mZckBdrIx0orEtZV
        4bmp/YzhwvcubU4=;
Received: from client1.football.example.com  [192.0.2.1]
      by submitserver.example.com with SUBMISSION;
      Fri, 11 Jul 2003 21:01:54 -0700 (PDT)
From: Joe SixPack <joe@football.example.com>
To: Suzie Q <suzie@shopping.example.net>
Subject: Is dinner ready?
Date: Fri, 11 Jul 2003 21:00:37 -0700 (PDT)
Message-ID: <20030712040037.46341.5F8J@football.example.com>

Hi.

We lost the game. Are you hungry yet?

Joe.
```

L'en-tête en question spécifie l'algorithme de chiffrement utilisé (ici SHA-256 et RSA), le domaine signant (ici `example.com`), un **sélecteur** (ici `brisbane`) qui servira à sélectionner la bonne clé, les en-têtes effectivement signés (ici `Received : From : To : Subject : Date : Message-ID`), la signature elle-même et l'**identité** de l'émetteur (ici `joe@football.example.com`). (La différence

entre le domaine, identifié par d= et l'« identité » indiquée par i= est subtile et a dû faire l'objet d'une mise à jour, dans le RFC 5672, mise à jour qui a depuis été intégrée dans la nouvelle norme DKIM, le RFC 6376.)

L'identité est indiquée dans la signature pour éviter les longs débats sur l'identité la plus pertinente parmi toutes celles présentes dans un message (cf. le RFC 4407 pour un des exemples d'algorithme qui croit naïvement qu'il existe une et une seule bonne identité). Notre RFC note que le MUA doit en tenir compte et doit afficher l'adresse qui est authentifiée, pas seulement celle qui se trouve dans le champ From et qui peut être différente.

Le programme qui veut vérifier la signature (en général un MTA mais cela peut être le MUA) va devoir récupérer la clé de signature. DKIM ne souhaitant pas dépendre des lourdes et chères autorités de certification, la clé est récupérée via le DNS (d'autres méthodes sont en cours de normalisation). Pour le message ci-dessus, la requête DNS sera brisbane._domainkey.example.com et un enregistrement DNS de type TXT contiendra la clé. Pour voir une clé réelle, vous pouvez taper dig TXT beta._domainkey.gmail.com..

Un des problèmes difficiles en cryptographie est que les messages sont souvent modifiés en cours de route. Par exemple, un logiciel imposé par le direction de l'entreprise va ajouter automatiquement un stupide message pseudo-légal comme « Ce message e-mail et les pièces jointes sont transmis à l'intention exclusive de ses destinataires et sont confidentiels et/ou soumis au secret professionnel. Si vous recevez ce message par erreur, merci de le détruire et d'en avertir immédiatement l'expéditeur par téléphone ou par mail. Toute utilisation de ce message non conforme à sa destination, toute diffusion ou toute publication, totale ou partielle, est interdite, sauf autorisation. L'intégrité de ce message n'étant pas assurée sur Internet, nous ne pouvons être tenu responsables de son contenu. » (notons que le dernier point devient faux avec DKIM). Ou bien une liste de diffusion va mettre des instructions de désabonnement. DKIM traite ces problèmes avec deux méthodes : une canonicalisation (section 3.4 du RFC) du message (plusieurs algorithmes sont disponibles) pour limiter les risques qu'une modification triviale ne fausse le calcul de la signature et l'option l= qui permet d'indiquer sur quelle distance le message est signé. Si on indique l=1000, seuls les mille premiers octets seront signés et une liste pourra ajouter un message automatiquement à la fin, cela n'invalidera pas la signature.

Malheureusement, DKIM encourage également (section 5.3) à remplacer les caractères codés sur 8 bits (comme l'UTF-8 ou le Latin-1) par des horreurs comme le Quoted-Printable, pour limiter les risques qu'une conversion automatique en Quoted-Printable, comme le font certains MTA, n'invalide la signature. Des années d'efforts pour faire passer les caractères 8-bits dans le courrier sont ainsi négligés.

Parmi les limites de DKIM (qui sont celles de beaucoup de solutions de sécurité, un domaine complexe où il faut faire beaucoup de compromis), il faut aussi se rappeler qu'un attaquant actif peut tout simplement retirer complètement une signature. Tant que les domaines ne publient pas leur politique de sécurité (en suivant le RFC 5617) et donc annoncent « Nous signons toujours », cette attaque passera inaperçue.

Aujourd'hui, tous les messages sortant de Gmail sont signés avec DKIM et des implémentations de DKIM existent pour plusieurs logiciels (comme le DKIM milter <http://sourceforge.net/projects/dkim-milter> de sendmail). Mais peu de domaines vérifient les signatures.

Enfin, notons que DKIM est une technologie d'authentification, pas d'autorisation. Cette distinction est cruciale <https://www.bortzmeyer.org/authentifier-et-autoriser.html>. DKIM peut prouver que le message vient bien de Nicolas.Sarkozy@elysee.fr, il ne peut pas dire si la personne en question est digne de confiance ou si elle va réellement se préoccuper de « la France qui se lève tôt ».