Tordons le cou au phishing

Damien Mascré, David Verdin, Laurent Aublet-Cuvelier

Tordons le cou au phishing

Damien Mascré
RENATER
23-25, rue Daviel
75013 Paris

David Verdin
RENATER
23-25, rue Daviel
75013 Paris

Laurent Aublet-Cuvelier
RENATER
23-25, rue Daviel
75013 Paris

Résumé

À la liste des nombreux maux dont souffre le mail, le phishing fait figure de vérole : rare, honteux, mais malheureusement dévastateur.

Et pourtant : SPF, DKIM, DMARC, etc. L'IETF ne manque pas de RFC pour améliorer le niveau de méfiance de la messagerie. Le but de ce mille-feuille de plusieurs centaines de pages est de combattre les messages illégitimes selon la méthode de la « méfiance par défaut ». En bref cela nous donne carte blanche pour rejeter des messages, même légitimes. La RFC ARC, introduisant la notion de confiance sélective, on peut améliorer encore la pertinence du filtrage.

L'objet de cet article est de vous (re)présenter ces RFC et la façon dont elles fonctionnent ensemble. Notre communauté peut vaincre le phishing si :

1. *elles sont implémentées sur une large échelle ;*
2. *nous faisons converger nos politiques de filtrage.*

Bien que nous ayons mené des expérimentations que nous souhaitons partager, cet article ne vous donnera pas de solution toute faite. Nous voulons que la collaboration au sein de la communauté commence ici, aux JRES. Tout le monde est là, c'est le moment de parler.

Mots-clefs

phishing, email, RFC, SPF, DKIM, DMARC, ARC, organisation

1. Introduction

Un phishing, ou « hameçonnage » en français, est un message dont le but est d'inciter le lecteur à accéder à des contenus, liés au message ou distants, qui permettront à l'initiateur de l'hameçonnage de lui voler des informations : identifiants de connexion, numéro de carte bleue, etc.

Il existe plusieurs types d'hameçonnages. Ceux qui nous intéressent sont les hameçonnages de masse, qui consistent en des messages génériques envoyés à des milliers, voire des millions d'adresses. Le contenu du message, lu par n'importe quel utilisateur à tête reposée, est en général dépourvu de toute crédibilité. Le succès de ces hameçonnages repose donc sur le fait que, statistiquement, s'il est envoyé à un grand nombre de destinataires, le message atteindra fatalement un utilisateur qui se trouvera exactement dans le bon état d'esprit pour tomber dans le piège. Fatigue, dépression, maladie, peur, coïncidence, sont autant de circonstances qui rendront les utilisateurs vulnérables aux pirates. Et qui, même s'il est informaticien chevronné, peut prétendre, sans se mentir à lui-même, ne jamais se trouver dans un de ces états au mauvais moment : celui où l'hameçonnage fatal atteint sa boîte mail ?

Pour ces hameçonnages de masse, un bon nombre de RFC (« Request For Comments », recommandations développées par l'Internet Engineering Task Force, ou IETF) devraient, si elles étaient appliquées, en réduire le nombre et l'impact.

Nous en présentons quatre qui forment un ensemble cohérent : SPF, DKIM, DMARC et ARC. Elles ont toutes pour objectif de valider la source d'un message.

L'objet de cet article est de présenter ces RFC, leur usage et la manière dont notre communauté peut s'emparer d'elles pour diminuer le nombre d'hameçonnages.

Remarque essentielle : qu'un message vienne d'une source valide n'offre aucune garantie sur le contenu du message lui-même. L'ensemble des mécanismes présentés ici ont donc vocation à compléter un dispositif anti-spam / anti-hameçonnage déjà présent.

2. SPF et DKIM : authentifier les serveurs

2.1 Pourquoi authentifier les serveurs ?

Enfonçons donc quelques portes ouvertes, en guise de rappel.

Lorsqu'un email est envoyé, le serveur expéditeur ouvre une session SMTP vers le serveur du destinataire en associant une adresse d'expédition à l'entrée « MAIL FROM » de la session. Cette adresse, communément appelée « From de l'enveloppe », n'est utilisée que le temps de la connexion entre les deux serveurs. Dans la section « DATA » de la session, le serveur expéditeur fournit l'ensemble du message, y compris les entêtes. Parmi ceux-ci se trouve l'entête « From ». C'est la valeur du champ « From » qui sera affichée dans le client courrier et qui déterminera donc, dans l'esprit du destinataire, l'identité de l'expéditeur.

Or, rien n'est plus facile que d'usurper un champ « From ».

Par exemple, avec le client Thunderbird, on peut définit quelle adresse d'expédition mettre dans le champ « From ».

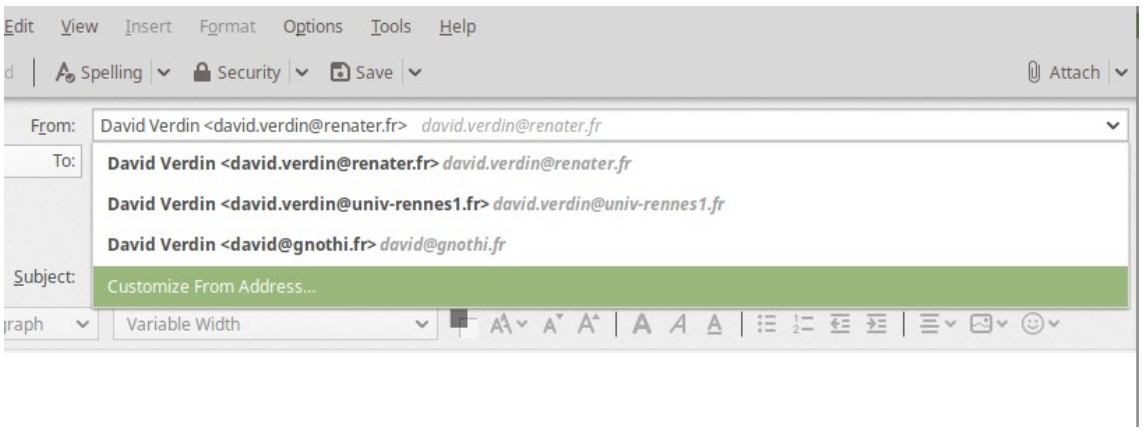


Figure 1: Option de Thunderbird pour saisir l'adresse d'expédition à positionner dans le champ « From ».

Une fois la session acceptée, le contenu du message, contenu par la section « DATA » peut recevoir n'importe quel contenu. On peut donc mettre n'importe quelle valeur au champ « From ». Le contenu sera transmis au destinataire défini dans la section « RCPT TO ». Tout serveur de mail réagira de même.

C'est ce mécanisme que nous illustrons dans la figure ci-dessous.

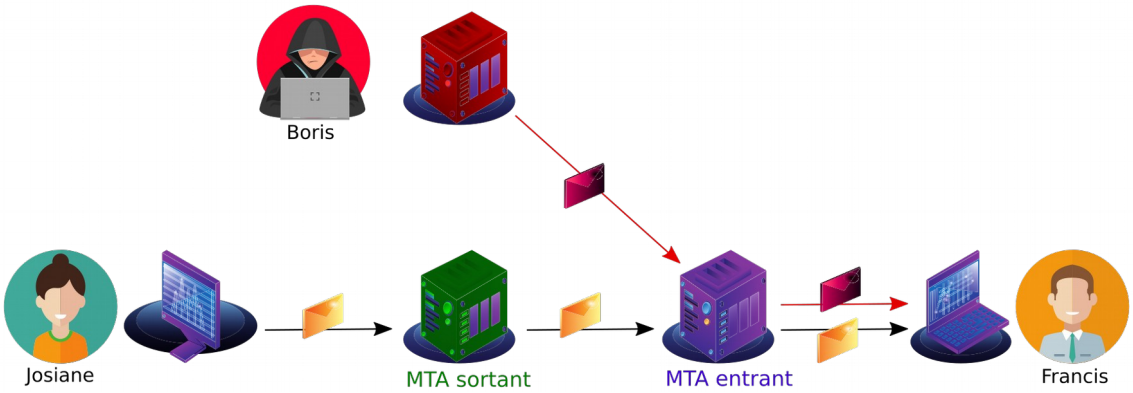


Figure 2: Le fonctionnement SMTP de base permet l'usurpation d'identité.

Dans les deux sessions, seule celle de Josiane est légitime. Celle de Boris est initiée depuis un autre serveur en s'appuyant sur la possibilité offerte par le protocole SMTP de saisir n'importe quel champ « From ». Le serveur représenté en vert est légitime, celui représenté en rouge ne l'est pas.

Dans le fonctionnement SMTP de base, rien n'empêche donc de prétendre être un domaine plutôt qu'un autre. L'authentification des serveurs a pour but de repérer les sources légitimes.

2.2 SPF

L'objet de la RFC SPF [1], pour « Sender Policy Framework », est d'identifier les hôtes autorisés à envoyer du mail au nom d'un domaine.

On comprend l'intérêt de la démarche : s'il est possible de distinguer les IP qui relèvent effectivement du domaine, il devient facile de repérer les serveurs émettant des mails de manière indue.

Cette RFC étant bien connue depuis longtemps, nous avons reporté le détail de son fonctionnement dans l'annexe 1 du présent article. Retenons juste le fonctionnement de base suivant (qui est le plus couramment rencontré).

Dans un enregistrement DNS de type TXT, le gestionnaire d'un domaine définit les adresses IP autorisées à diffuser des messages au nom de ce domaine.

Lorsqu'un serveur de messagerie reçoit un message, il consulte l'enregistrement SPF du domaine et évalue la légitimité du serveur émetteur.

Le résultat de l'évaluation est stocké soit dans un entête « Received-SPF », soit dans un entête « Authentication-Results », soit les deux.

Par exemple, on peut trouver ces entêtes après un SPF évalué à « pass » :

```
Received-SPF: pass (destination.tld: domain of
utilisateur@origine.tld designates 192.168.0.0 as permitted
sender) client-ip=192.168.0.0;
Authentication-Results: mx.destination.tld;
       spf=pass (destination.tld: domain of
utilisateur@origine.tld designates 192.168.0.0 as permitted
sender) smtp.mailfrom=utilisateur@origine.tld;
```

À partir de là, c'est au serveur destinataire d'appliquer un traitement adéquat au message en fonction de la valeur positionnée dans les entêtes SPF.

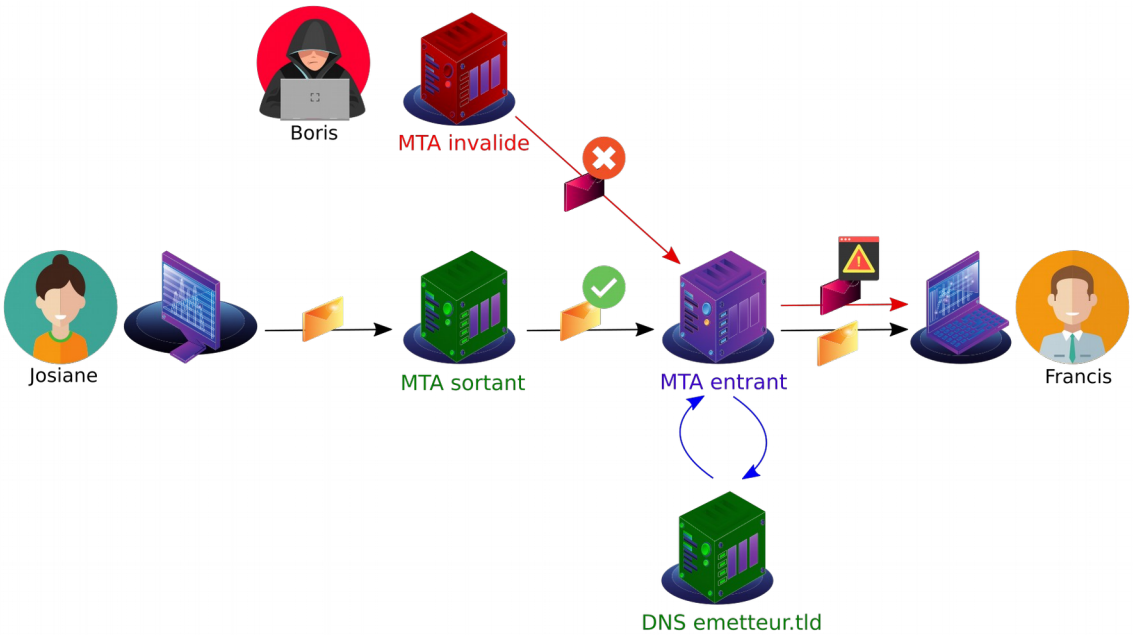


Figure 3: SPF permet de valider la source d'un message

Dans la plupart des cas, c'est le logiciel antispam qui exploitera le résultat de l'évaluation SPF en modifiant la note globale de spam du message. Ceci dit, les valeurs attribuées pour échec de validation SPF sont par défaut trop faibles pour faire fortement pencher la balance vers un classement en spam.

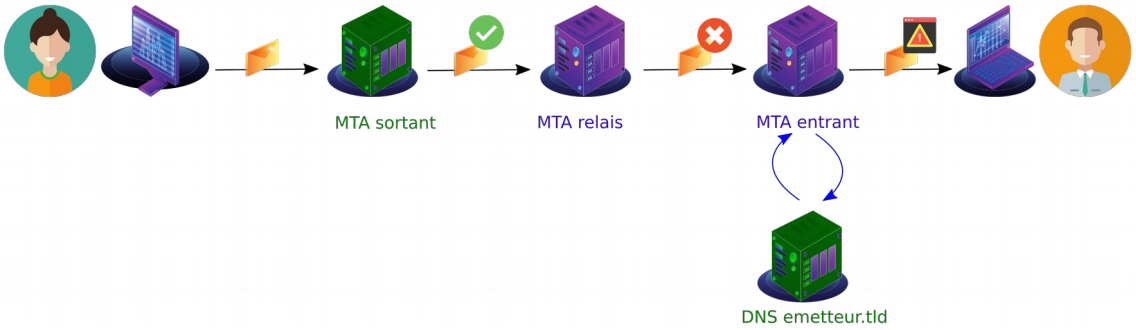


Figure 4: SPF n'est utilisable que lors de la session avec le SMTP sortante. un relais rend impossible sa validation.

Dans les cas où des services font suivre les mails, comme lorsque l'on positionne une redirection dans une boîte mail ou lorsqu'on utilise des listes de diffusion, SPF sera violé systématiquement. En effet les serveurs émetteurs de ces services ne sont pas, en général, inclus dans l'enregistrement SPF de l'expéditeur. Ces systèmes contournent ce problème en peuplant l'entrée « MAIL FROM » de leur session sortante par une adresse locale (l'adresse de la liste ou une adresse réécrite par SRS). Ils positionnent eux-même un enregistrement SPF pour leur propre domaine, ce qui leur permet de réussir la validation SPF, non pas avec l'enregistrement de l'expéditeur mais avec le leur.

Conclusion :

– SPF permet de valider le serveur source du message ;

– un serveur intermédiaire (d'un autre domaine) peut disposer de son propre enregistrement SPF et réécrire le « MAIL FROM » pour réussir la validation qui, sinon, échouerait ;

– en cas de non respect de SPF, c'est au serveur de destinataire de choisir la politique à appliquer.

2.3 DKIM

DKIM [2] (« Domainkeys Identified Mail ») permet à un serveur émetteur d'endosser une certaine responsabilité vis à vis d'un message en reliant celui-ci à son domaine d'expédition par le biais d'une signature.

Ce protocole étant bien connu, nous avons repoussé sa description détaillée dans l'annexe 2. Nous ne présentons ici que son principe. Reportez-vous à l'annexe s'il vous manque des informations.

2.3.1 Pourquoi signer ?

Avant d'arriver à son destinataire, un message peut passer par plusieurs intermédiaires susceptibles de l'altérer. L'arrivée d'un message avec une signature valide permet de s'assurer que les éléments signés n'ont pas été modifiés.

Contrairement à S/MIME, qui ne signe que le corps du message, DKIM inclut une partie des entêtes (dont le champ « From ») dans la signature. Les entêtes ne peuvent donc pas être altérés sans casser la signature.

C'est une bonne chose car, comme il n'y a pas de lien direct entre l'identité de l'émetteur et la clé employée pour signer le message, rien n'empêcherait que les éléments liés à cette identité (comme le champ « From » du message) soient altérés en cours de route. La signature des entêtes est la parade à ce problème.

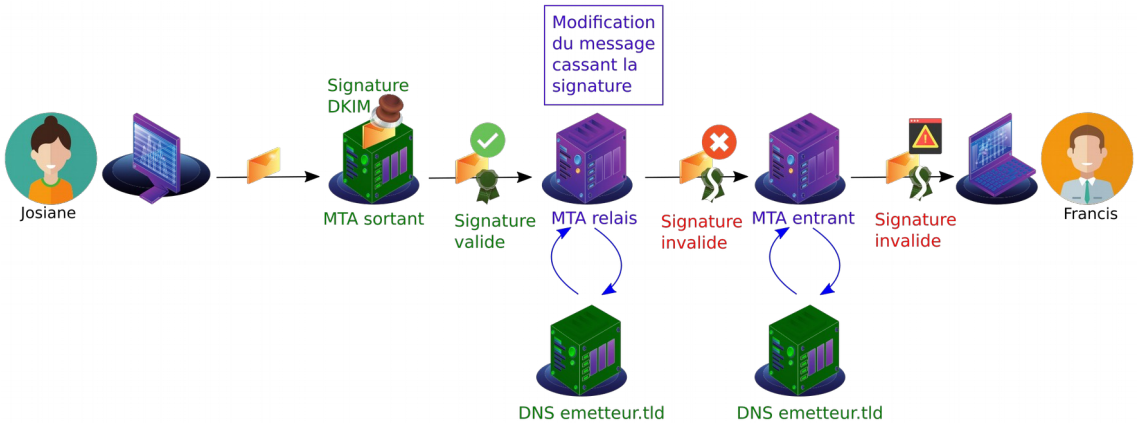


Figure 5: DKIM permet de vérifier l'intégrité d'un message grâce à la pose d'une signature.

DKIM résout le problème de SPF vis à vis des relais. En effet, tant que le message n'est pas modifié, la signature reste valide, quelque soit le nombre de relais intermédiaires entre le serveur qui a apposé la signature DKIM et celui qui va évaluer la validité de cette signature.

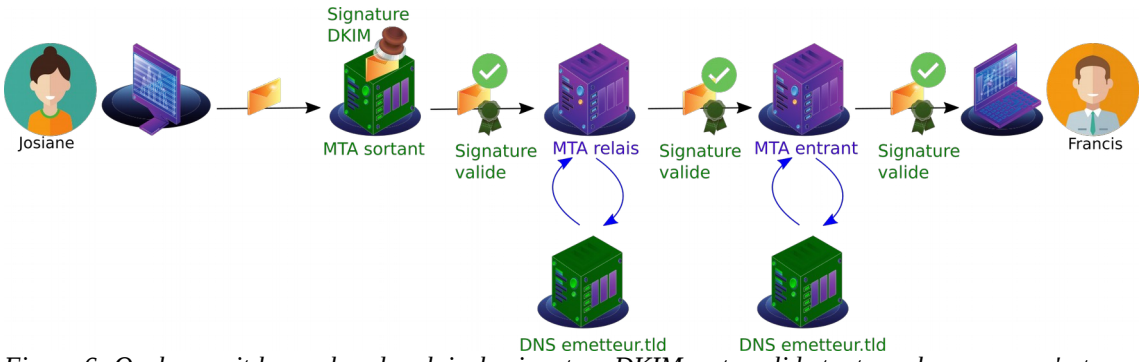


Figure 6: Quelque soit le nombre de relais, la signature DKIM reste valide tant que le message n'est pas altéré.

2.3.2 Et après ?

Tout d'abord, l'absence de signature est un bruit blanc. Entendez : cela ne permet pas de valider quoi que ce soit. La présence d'un enregistrement n'engage en effet aucunement l'expéditeur à signer tous ses messages. L'absence de signature ne veut pas dire que le message est illégitime. A contrario, la présence d'une signature ne signifie pas que le message n'est pas un spam ou un hameçonnage.

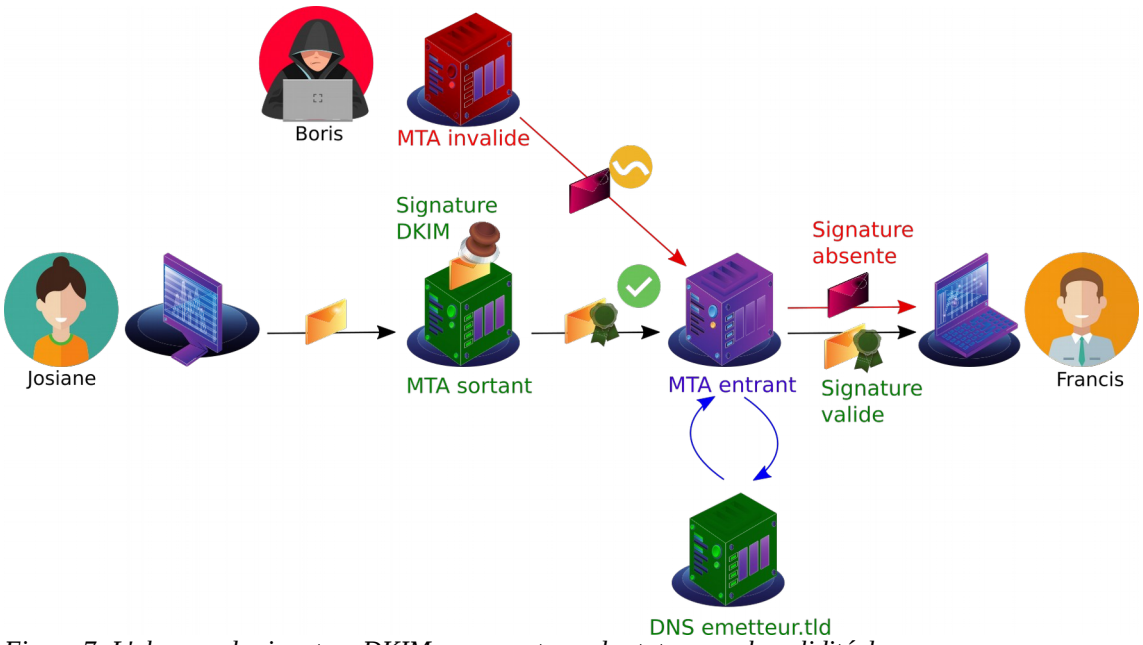


Figure 7: L'absence de signature DKIM ne permet pas de statuer sur la validité du message.

Une fois la signature validée, c'est au destinataire de décider quoi faire. Souvent, le résultat de la validation de la signature est stocké dans un entête « Authentication-Results », comme pour SPF.

Ce comportement n'est pas systématique et, tout comme pour SPF, le traitement ultérieur du message n'est pas normalisé, que la signature soit valide ou non.

Conclusion :

– DKIM permet de vérifier l'intégrité du message et d'authentifier sa source ;

– l’absence de signature DKIM n’est pas un indice de malversation ;

– en cas de signature DKIM invalide, c’est au serveur de destinataire de choisir la politique à appliquer.

2.4 Bilan sur SPF et DKIM

On dispose, avec SPF et DKIM, de deux méthodes permettant d’authentifier la source d’un message, soit comme émetteur de message valide (SPF), soit comme détenteur de la clé privée ayant signé le message (DKIM) ; l’intégrité de la signature du message garantit dans une certaine mesure que celui-ci n’a pas été altéré en route.

Mais le traitement à faire subir au message n’est jamais indiqué. La décision reste à la charge du destinataire.

Dans beaucoup de serveurs de messagerie, le service anti-spam placé en frontal ajuste sa note en fonction de la conformité aux RFC. Néanmoins, cela n’est ni généralisé, ni homogène.

On se trouve donc dans la situation paradoxale de serveurs émetteurs faisant de leur mieux pour prouver leur bonne foi, sans pour autant disposer d’un traitement homogène en bout de chaîne par les serveurs destinataires.

DKIM est également facile à contourner puisqu’une signature n’est jamais obligatoire. Un tiers peut donc modifier le message et supprimer la signature. Les serveurs de listes ont ainsi contourné la question DKIM en remplaçant la signature d’un expéditeur par une signature du serveur de listes. De cette manière, une signature DKIM restait présente dans le message, mais signée par une clé privée du serveur de listes.

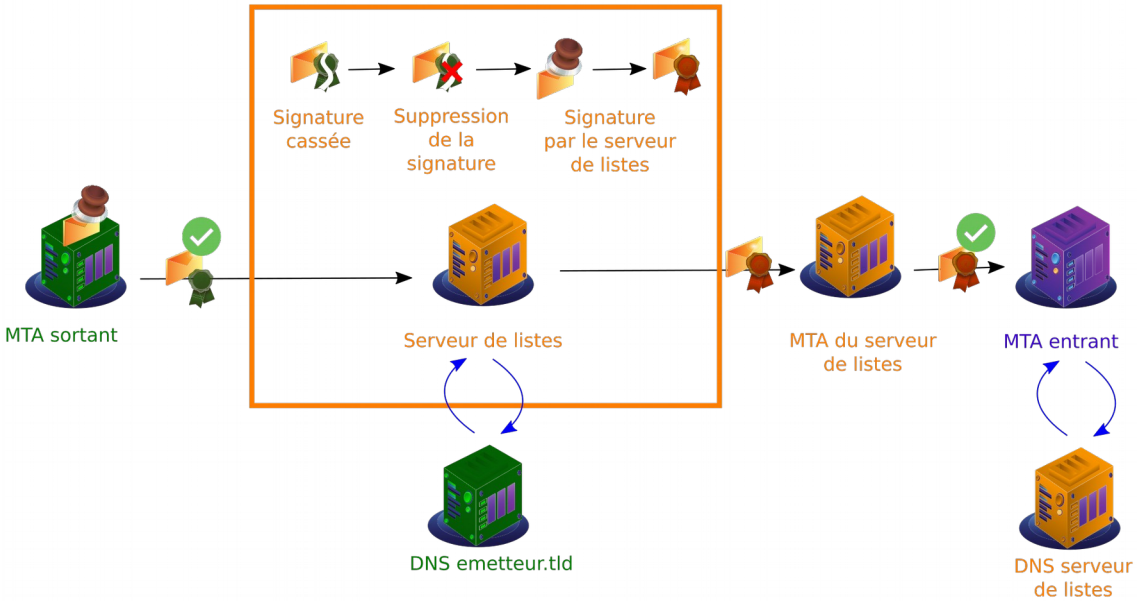


Figure 8: Un serveur de listes remplace une signature DKIM cassée par sa propre signature.

Une autre limite à ces deux protocoles est la distance sémantique entre leur validité et l’interprétation qui peut en être faite par l’utilisateur destinataire du message. Ainsi, la présence d’une signature DKIM valide dans un message ne signifie pas que cette signature émane de l’expéditeur du message identifié dans le champ « From ». Par

exemple, si un serveur de listes envoie un message en ajoutant sa signature DKIM, la seule chose dont on peut être sûr, c'est que c'est bien ce serveur qui l'a émis et que le message est tel qu'il l'a envoyé. Mais nous n'avons aucune garantie qu'il n'a pas été altéré avant d'accéder au serveur de listes.

On peut donc leur reprocher de laisser trop de latitude au serveur destinataire et de ne pas permettre de certifier que l'expéditeur du message est bien celui du champ « From ».

Tout ceci a changé avec DMARC.

3. DMARC : établir une politique de sécurité

DMARC [3] pour « Domain-based Message Authentication, Reporting and Conformance », est une RFC dont le seul objectif et de permettre à un serveur émetteur de définir une politique de sécurité. En bref, il s'agit de dire aux serveurs destinataires ce qu'il convient de faire si l'authentification du serveur émetteur, du point de vue de SPF et de DKIM, échoue.

3.1 Pourquoi une politique ?

Bien que SPF et DKIM offrent des mécanismes solides d'authentification des serveurs, on constate que le traitement en bout de chaîne est trop hétérogène pour s'appuyer de manière fiable sur ces RFC pour la sécurité de la messagerie.

3.2 Principe de DMARC

Brièvement, un enregistrement DMARC contient la politique ainsi que les moyens d'envoyer des rapports au gestionnaire du domaine. Les détails concernant la description de cette politique sont exprimés dans un enregistrement DNS, dont vous trouverez la description en annexe 3.

3.2.1 Authentication (Domain-based)

L'authentification, dans DMARC, repose sur SPF et DKIM. Si l'un ou l'autre de ces mécanismes est évalué à « pass », le traitement du message continue comme si de rien n'était. Si aucun des deux n'est validé, on applique la politique définie dans l'enregistrement DMARC.

Mais seulement s'il y a alignement.

Dans le cadre de DMARC les vérifications SPF et DKIM sont faites en se fondant sur le domaine indiqué dans le champ « From ». Toute signature DKIM émise depuis un autre domaine, ou tout serveur émetteur absent du SPF de ce domaine seront ignorés.

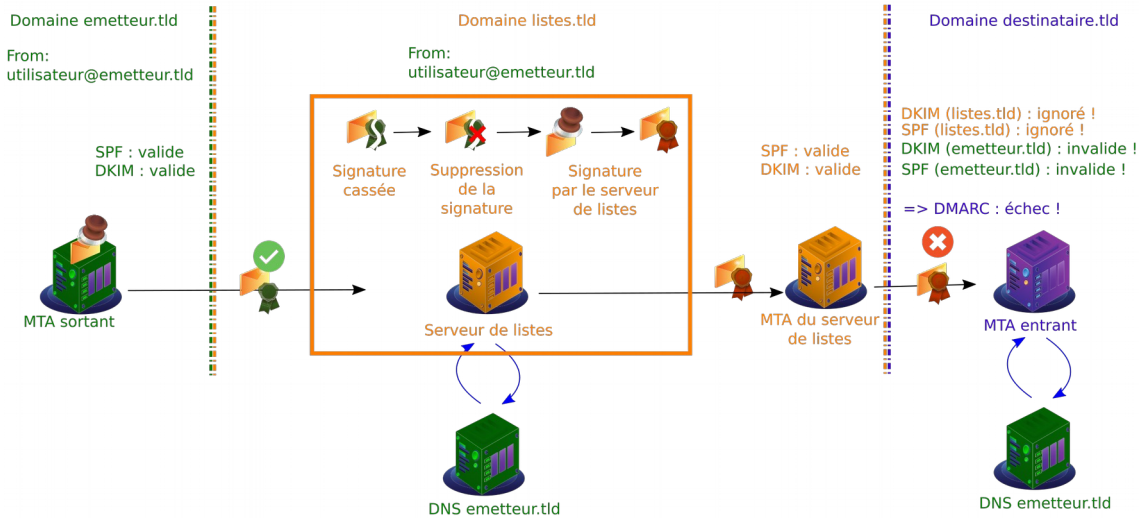


Figure 9: Un serveur de listes remplaçant une signature DKIM cassée par la sienne ne passera pas la politique DMARC.

3.2.2 Reporting

DMARC prévoit l'envoi de rapports, par défaut une fois par jour, sur les messages reçus par les serveurs l'implémentant.

Un des intérêts de l'existence de tels rapports est d'être averti en cas de souci de configuration et d'utilisation malicieuse du nom de domaine.

3.2.3 Conformance

La conformité (« conformance » en anglais) est le respect de la politique définie par l'enregistrement DMARC : s'il indique « p=reject », le message doit être rejeté si l'évaluation DMARC a la valeur « fail ». Rappel : l'authentification DMARC a la valeur « pass » si et seulement si au moins l'un des deux mécanismes de base, SPF ou DKIM, a été évalué avec la valeur « pass ».

Par exemple, supposons que le domaine émetteur ait positionné les deux enregistrements suivants :

```
emetteur.tld.         86400  IN     TXT      "v=spf1 a mx -all"
_dmarc.emetteur.tld. 86400  IN     TXT      "v=DMARC1; p=reject;
rua=mailto:abuse@emetteur.tld"
```

Seuls les serveurs A et MX du domaine « emetteur.tld » sont autorisés à émettre des messages. Tout autre origine donne un résultat « fail » pour l'évaluation SPF.

L'enregistrement DMARC positionne une politique « p=reject ». Tout message positionnant un champ « From : » du domaine « emetteur.tld » venant d'un autre origine que les serveurs A ou MX et sans signature DKIM (ou avec une signature invalide) devra être rejeté en session par le destinataire.

On peut définir trois politiques :

– « reject »,

– « quarantine »,

– « none ».

La politique « reject » est très claire : le message est rejeté en session. Un avis de non remise du message est envoyé à l'expéditeur, en général avec le code d'erreur 5.7.1.

La politique « quarantine » est plus complexe et plus floue. Très peu d'entités l'emploient. Le problème ici est que la quarantaine est inégalement implémentée suivant les acteurs. On retourne un peu sur le flou précédant l'introduction de DMARC.

La politique « none » revient à ne rien exiger des destinataires du message. Cependant, le message reçu contiendra l'entête « Authentication-Results » avec une valeur « fail » pour l'authentification DMARC, tout comme pour SPF et DKIM. Ceci peut suffire pour que le classement en pourriel soit déclenché. Cette valeur est également très utile pour tester DMARC, en activant l'envoi de rapports dans l'enregistrement à des fins d'analyse.

3.3 Limites de DMARC

Dès lors qu'un enregistrement DMARC positionne une politique à « p=reject », le nombre de possibilités d'usurpation d'identité diminue drastiquement. C'est donc un système particulièrement efficace, dès lors :

– qu'il est supporté par les MTA destinataires ;

– que le mail ne passe pas par des mécanismes de renvoi de mail, tels que des redirections dans les boîtes mail ou des listes de diffusion.

C'est d'ailleurs le principal reproche que l'on peut faire à ce mécanisme. Il existe des procédés légitimes dans la messagerie qui cassent SPF ou DKIM, ou les deux, et donc produire un résultat « fail » pour DMARC. C'est ce qui se produit pour les listes de diffusion, qui ont dû mettre en place des mécanismes de contournement pour que les messages soient diffusés.

Concrètement, quand une politique DMARC jugée trop agressive existe pour le domaine d'expédition d'un mail à destination d'une liste de diffusion, les serveurs remplacent le contenu du champ « From » par une autre adresse, par exemple celle de la liste.

```
From: Inigo Montoya (via sympa-users Mailing List) <sympa-
users@listes.renater.fr>
```

Exemple de champ « From » modifié par un serveur de listes. L'adresse email de l'expéditeur est remplacée par celle de la liste et son identité apparaît seulement dans le gecos.

De ce fait, les messages sont bien diffusés, mais il est difficile de comprendre, pour l'utilisateur destinataire, qui est l'auteur du message. En outre, la disparition de l'adresse d'expédition empêche d'utiliser la fonction « Répondre » des clients de messagerie.

En conclusion, DMARC c'est très bien, mais l'utilisation des redirections est impossible avec cette solution d'authentification. Or la redirection est un mécanisme très important en messagerie. Toute solution d'authentification la rendant impossible est donc caduque.

C’est pour cela qu’une amorce de cercle de confiance a été proposée dans le cadre de la RFC ARC.

4. ARC : mettre en place la confiance

ARC [4] pour « Authenticated Received Chain », est une RFC visant à mettre en place une chaîne de responsabilité dans la messagerie.

L’intérêt de cette chaîne de responsabilité est de permettre qu’un message émanant d’une source de confiance mais dont un intermédiaire va violer la politique DMARC soit tout de même accepté. Ceci permet, par exemple, à un serveur de listes de ne pas avoir à réécrire le champ « From » d’un message pour qu’il soit accepté.

4.1 Principe d’ARC

Le principe d’ARC est le suivant : lors du trajet d’un mail, depuis le serveur émetteur jusqu’au serveur destinataire, chaque tiers peut sceller le message avant de le transmettre au relais suivant. Chaque sceau renferme les informations d’authentification (le résultat des vérifications stocké dans Authentication-Results) au moment de la réception du message par le tiers. L’ensemble de ces sceaux forme la chaîne d’authentification (« Authenticated Received Chain » en anglais) qui donne son nom à la RFC.

La description et des exemples d’entêtes peuvent être trouvés dans l’annexe 4.

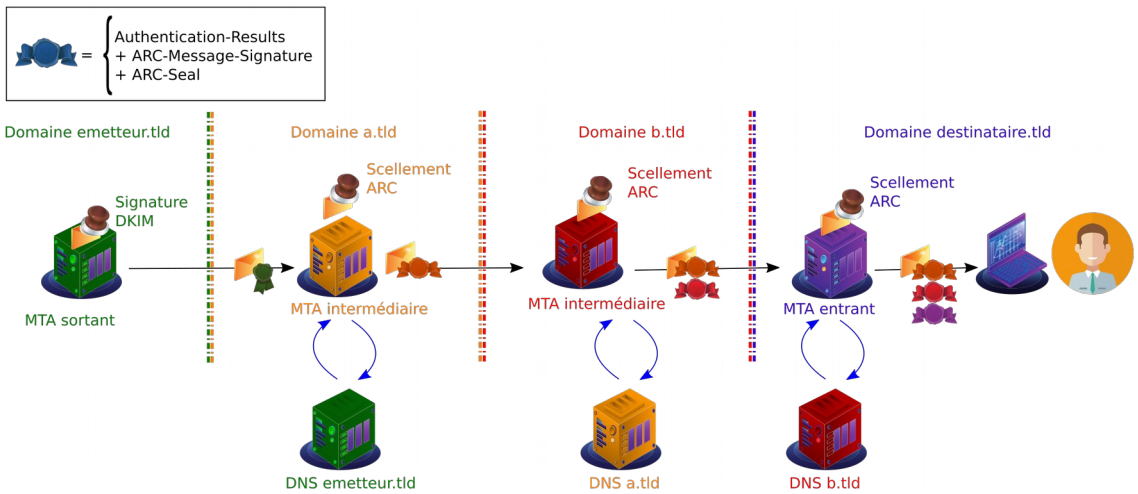


Figure 10: Dans le cadre d'ARC, tout serveur intermédiaire peut apposer son sceau et prolonger ainsi la chaîne d'authentification.

Ainsi, à chaque étape où l’authentification est valide, on peut définir un nouveau résultat dans « Authentication-Results » : « arc » qui pourra être évalué par le destinataire.

4.2 Et après ?

À ce stade, le lecteur se demande : « c’est bien joli tout ça, mais à part rajouter un couche d’entêtes dans nos messages, ça sert à quoi ? »

Une impatience bien compréhensible après avoir lu toutes ces pages de description de RFC.

Eh bien, lecteur, la nouvelle méthode d'authentification, « arc », permet d'établir un compromis avec les autres. En termes simples, un serveur destinataire peut choisir de positionner l'authentification « arc » sur la valeur « pass » (ce qui revient à accorder sa confiance au message) *même si toutes les autres méthodes ont la valeur « fail »*.

Pour comprendre cela, il faut préciser ici comment un MTA supportant ARC vérifie le message.

4.3 Évaluation ARC

Supposons que le message échoue à son authentification DMARC et que la politique soit « p=reject ». Le serveur vérificateur devrait le rejeter en session. Mais si le contenu de l'entête « ARC-Authentication-Results » du triplet apposé par le prédécesseur avait une valeur « pass » et que le vérificateur fasse confiance à ce serveur, alors il peut inscrire la clé « arc=pass » dans son propre entête « ARC-Authentication-Results ». L'authentification sera alors considérée comme réussie.

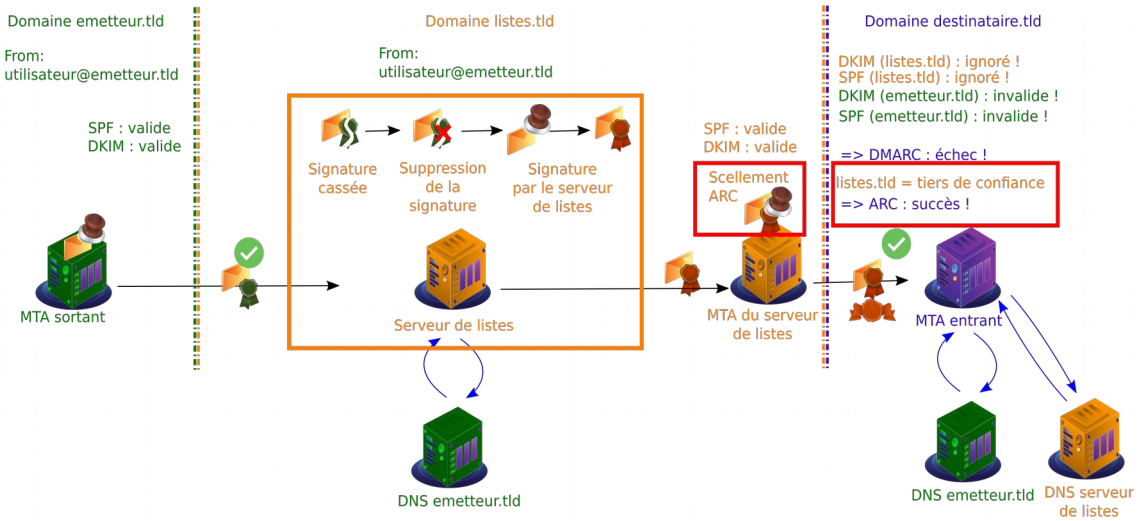


Figure 11: L'authentification serveur d'un serveur de listes peut être considérée valide par le destinataire s'il appose son sceau ARC et qu'il est recensé comme tiers de confiance.

Tout le problème est maintenant de savoir comment, concrètement, mettre en place cette confiance.

5. Une organisation anti-hameçonnage autour de ces RFC

Parvenus à ce point, il semble pertinent de résumer où nous en sommes.

Nous disposons donc de :

- SPF et DKIM pour authentifier les sources des messages ;
- DMARC pour définir ce qu'il doit arriver aux messages ne respectant ni SPF, ni DKIM ;
- ARC pour permettre de laisser passer les messages ne validant pas DMARC mais ayant été validés en chemin par une source de confiance.

Pour atteindre l'objectif de cet article, à savoir se débarrasser de l'hameçonnage, il est nécessaire que le plus grand nombre de serveurs possible implémente SPF, DKIM et DMARC.

Pour mettre en place une politique stricte, il faut :

- implémenter DMARC avec un politique « reject » ;
- terminer les enregistrements SPF par « -all » ;
- Signer DKIM en sortie.

De cette manière, on s'assure la plus grande sévérité possible vis à vis des messages entrants. Évidemment, tout cela ne fonctionne que si les serveurs destinataires respectent cette politique.

Par ailleurs, une politique stricte vis à vis des sous-domaines de messagerie peut être envisagée. Si on sait quels domaines expédient des mails, il est possible de restreindre la possibilité d'employer des sous-domaines :

- en positionnant la paire « t=s » dans l'enregistrement DKIM (voir l'annexe 2) ;
- en imposant un alignement strict dans DMARC, via les paires « adkim=strict » et « aspf=strict » (voir l'annexe 4).

Ceci aura pour effet de réduire fortement le nombre d'hameçonnages arrivant à destination.

Cette proposition se heurte à deux difficultés :

1. certains serveurs peuvent envoyer des messages sans passer par un MTA sortant identifié ;
2. nombre de services légitimes cassent SPF et DKIM ;

Le passage à un tel niveau de rigueur impose donc d'une part une période de transition et d'analyse et, d'autre part, une organisation afin d'implémenter ARC.

Pendant la période de transition, on positionnera une politique DMARC à « p=none ». de cette manière, aucun message n'est rejeté et on peut analyser les rapports reçus. En particuliers, les rapports d'erreur permettront de repérer les serveurs hors norme de l'infrastructure.

D'un point de vue organisation, on peut agir sur deux leviers :

1. organiser une liste de serveurs de relais ;
2. collaborer pour améliorer l'implémentation.

Une liste de domaines intermédiaires de confiance pouvant modifier les messages (redirection ou listes) pourrait être centralisée et téléchargeable pour usage par les outils d'analyse ARC , à la manière des métadonnées d'une fédération d'identités.

Les contraintes envisageables à l'établissement d'une telle liste sont :

- sa vérification : s'assurer que les informations qui s'y trouvent sont fiables et à jour ;
- sa sécurisation : la rendre disponible de manière sécurisée.

Une fois les RFC implémentées, il faut rendre habituel l'affichage d'informations de légitimité (à l'instar du cadenas dans le navigateur). Notons qu'aucun client de messagerie générique ne propose d'affichage d'informations relatives à l'authentification de la source d'un message.

La difficulté sera également de réunir un nombre suffisant d'entités implémentant toutes ces RFC. La cible de cet article est la population des établissements d'enseignement supérieur et de recherche. Mais l'objectif est de faire tâche d'huile.

En résumé, l'intérêt de ces RFC est de forcer les serveurs de messagerie à montrer patte blanche, à l'instar de ce qui s'est passé pour le web quand la majorité des sites sont passés en HTTPS. À partir de là, on peut commencer à être volontairement sévère avec les domaines n'implémentant pas ces RFC. En clair, si on veut que les messages passent, il faut faire un effort. Ceci donnerait aux pirates l'obligation de les implémenter également. Et à partir de là, on a des garanties sur le domaine expéditeur, pour un message légitime comme pour un hameçonnage. Ayant des garanties sur le domaine, on peut légitimement recenser les domaines à problèmes, potentiellement contrôlés par des pirates, et ainsi enrichir des listes de serveurs de confiance ou de défiance.

6. Expérimentations à ce jour

6.1 État de l'art logiciel

Aujourd'hui il existe plusieurs implémentations pour vérifier SPF et DKIM, certaines sous forme de milter, soient parfois complètement intégrées au MTA, ou encore comme étape dans des suites intégrées (amavis, spamassassin, etc.). Parfois l'implémentation milter et celle des suites intégrées est issue du même code.

Les implémentations DMARC sont rares et les organisations qui implémentent la vérification DMARC en entrée le sont tout autant, ceci expliquant cela. Enfin les implémentations ARC se comptent sur les doigts d'une main. De Django Reinhardt.

Le tableau ci-dessous résume les fonctionnalités offertes par les implémentations existantes. Les bibliothèques sont signalées par un « (B) », les logiciels par un « (L) ». Les chiffres indiqués entre parenthèses renvoient aux remarques après le tableau.

	Vérifie SPF	Vérifie DKIM	Signe DKIM	Vérifie DMARC	Scelle ARC	Vérifie ARC	ARC permet d'ignorer DMARC
pyspf (B)	✔						
dkimpy (B)		✔	✔				
Mail::SPF (B)	✔						
Mail::DKIM (B)		✔	✔				
Mail::DKIM::ARC (B)					✔	✔	
Mail::DMARC (B)				✔			
libspf2 (B)	✔						
OpenDKIM (L)		✔	✔				
OpenDMARC (L)	(3)	✔		✔		(4)	(4)
OpenARC (L)					✔	✔	
authheaders (L)	✔	✔		✔			
Amavis (L) (1)	✔	✔	✔				
SpamAssassin (L)	(2)	(2)	(2)	(2)	(2)	(2)	(2)
authentication_milter (L)	✔	✔		✔	✔	✔	✔
Rspamd (L) (5)	✔	✔	✔	✔	✔	✔	(6)

Remarques :

1 la suite amavis, n'a pas été mise à jour depuis l'apparition de DMARC ;

2 la suite SpamAssassin possède un plugin qui lit les Authentication-Results laissés par les filtres individuels et permet d'affiner la notation du message ;

3 l'implémentation de la vérification SPF par OpenDMARC est réputée faillible et il est préférable de le configurer pour s'appuyer sur le résultat d'un autre filtre SPF ;

4 la version d'OpenDMARC qui incorpore la prise en compte d'ARC, la 1.4.x n'est pas encore disponible dans les distributions actuelles, il faudra la compiler soi même ;

5 Voir la présentation « Rspamd : logiciel anti-spam opensource, performant, évolutif et personnalisable » dans cette même édition des JRES pour en apprendre plus sur Rspamd ;

6 Il est peut-être possible d'écrire un filtre en Lua pour Rspamd qui s'appuie sur les résultats des vérifications ARC et DMARC.

Ainsi pour les expérimentations, il est préférable d'utiliser en entrée, au choix :

- authentication_milter ;
- la collection libspf2 / OpenDKIM / OpenARC / OpenDMARC (bonus : ils sont implémentés en C).

Et en sortie, pour signer et sceller :

- Rspamd (avec les modules DKIM et ARC) ;
- la collection OpenDKIM + OpenARC.

6.2 Et les autres ?

En septembre 2019, seuls Gmail et AOL, implémentent toutes ces RFC et utilisent des listes blanches sur réputations. Si une signature ARC ou DKIM est invalide, le message est automatiquement classé en spam. Ils scellent également les messages avec ARC.

Yahoo implémente DMARC, mais pas ARC. C'est le premier fournisseur à avoir positionné une politique DMARc à « p=reject ».

Fastmail évalue ces RFC et s'en sert pour influer sur la note de spam. Il est prévu que la politique DMARC évolue pour être positionnée à « p=reject ».

Les serveurs Microsoft scellent ARC mais ne font pas d'évaluation en entrée.

Beaucoup de prestataires, comme Free ou laposte.net ne tiennent aucun compte de ces RFC.

La situation, dans le privé, est donc très hétérogène. On observe cependant un durcissement progressif, au fur et à mesure de la diffusion de ces RFC.

6.3 Implémentation expérimentale chez RENATER

La mise en place d'une suite de filtres en réception ne pose pas de problème particulier : on applique chaque filtre l'un après l'autre et si DMARC = reject :

1. si l'un des domaines signataires ARC fait partie de notre liste blanche, on ignore le résultat DMARC ;
2. sinon on l'applique.

Si la décision est de laisser passer le message, on applique le reste (antispam, antivirus, etc.).

Il est plus intéressant de voir comment cela se comporte lorsqu'un message à notre destination sera modifié, par exemple parce nous hébergeons des serveurs de liste. Le chemin du message est présenté dans la figure 11, ci-dessous.

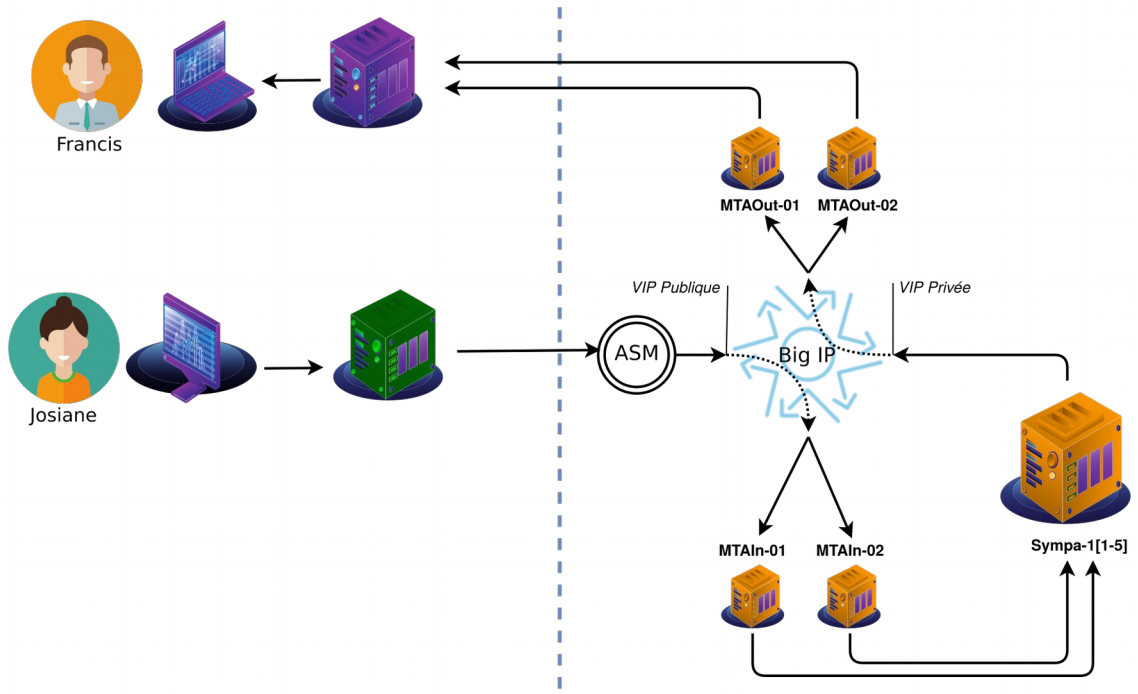


Figure 12: Dispositif expérimental à RENATER

Dans cette configuration, l'anti-spam mutualisé RENATER n'effectue pas de vérifications SPF et DKIM. La vérification SPF sur les machines MTAIn échouera donc systématiquement. Mais on a juste besoin d'une validation DKIM et d'un alignement correct pour valider DMARC et sceller ARC avant de transmettre le message au serveur de listes. Ce dernier modifie le message en altérant les entêtes « From » et « Subject » puis transfère le message aux machines de sortie MTAOut. Ces dernières apposent une signature DKIM et un scellé ARC puis transmettent le message aux domaines de destinations. Nous avons reproduit dans l'annexe 5 un message ayant traversé cette infrastructure, ainsi qu'une explication de texte sur les entêtes positionnés.

Par exemple, le message ci-dessous présente de bonnes garanties :

Message d'origine

ID du message	<4c5862ef-53af-bd4d-08cd-d2d1852b7e0e[illegible]>
Date de création :	13 septembre 2019 à 11:42 (Temps d'envoi : 73 secondes)
De :	David Verdin <[illegible]>
À :	testdmagp2@groupware2.renater.fr
Objet :	Re: [testdmagp2] test depuis p13 (spf + dkim, no dmarc)
SPF :	PASS avec IP 194.57.4.214 En savoir plus
DKIM :	'PASS' avec le domaine [illegible] En savoir plus
DMARC :	'PASS' En savoir plus

Figure 13: Résultats d'authentification d'un message valide affichés par Gmail.

Il sera tout de même classé dans le répertoire Spam et un avertissement sera affiché :

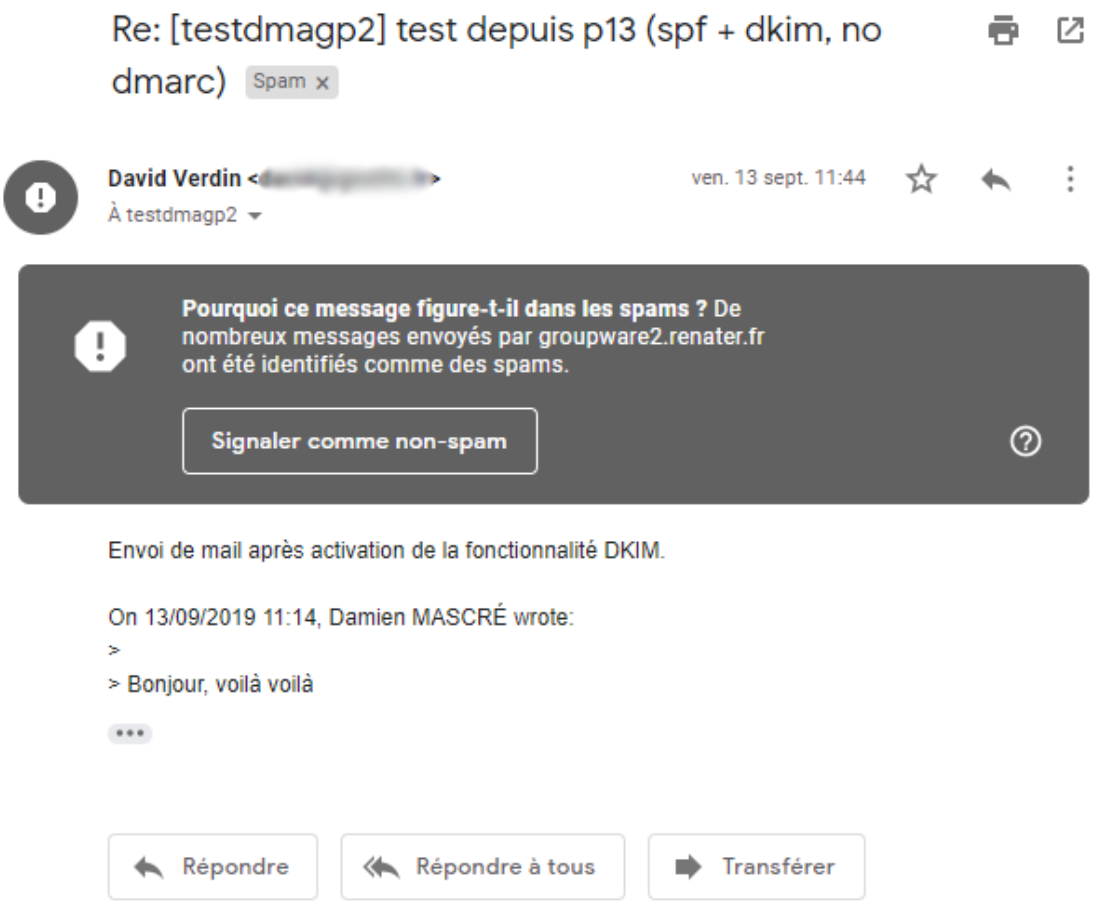


Figure 14: Un message valide mais tout de même catégorisé en spam par Gmail.

Dans le cas où notre message ne présente pas tous les gages, comme dans le cas d'une politique DMARC qui échoue, l'avertissement sera plus explicite :

Figure 15: Résultats d'authentification d'un message invalide affichés par Gmail.

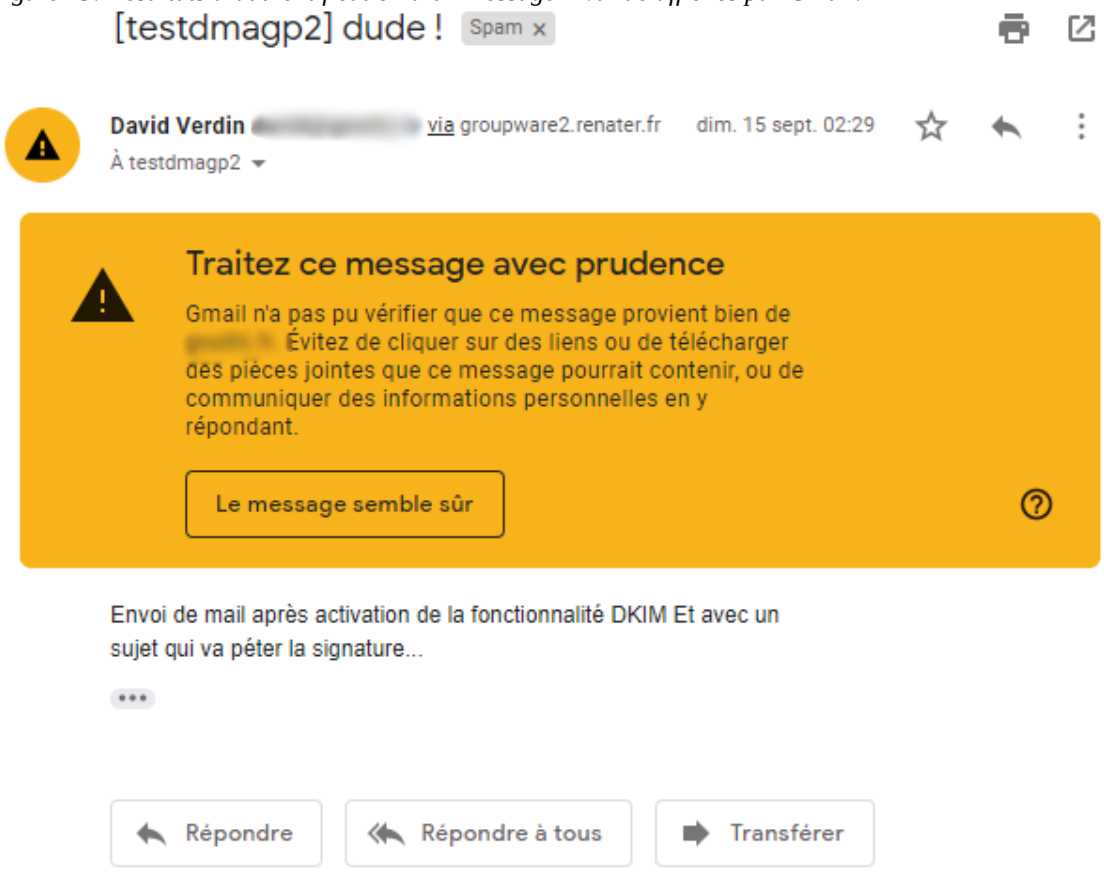


Figure 16: Avertissement affiché par Gmail pour un message dont l'authentification serveur a échoué.

7. Limites et propositions

Évidemment, même si nous sommes optimistes sur les possibilités offertes par ces RFC, nous sommes également conscient qu'elles ne représentent pas une panacée.

Une première limite est l'emploi de domaines non protégés. Par exemple, « le domaine presque comme le vrai ». Envoyez un message de la part du domaine rennater.fr et un certain nombre d'utilisateurs ne remarqueront que trop tard le « n » surnuméraire. Ou encore retaner.fr que le cerveau transformera automatiquement et inconsciemment en renater.fr lors de la lecture.

Évidemment, tous serveur implémentant ces RFC passerait à travers les mailles du filet.

D'ailleurs, regardez les hameçonnages que vous recevez ces derniers temps. Un certain nombre disposent d'une signature DKIM et d'une validation SPF on ne peut plus valides.

De la même manière, ce dispositif ne protège aucunement contre l'envoi depuis un compte légitime, mais piraté. Mais nous espérons que l'implémentation de ces RFC limitera la possibilité de pirater des comptes et réduira donc ce problème à la source.

Alors à quoi bon ?

Tout d'abord, ce n'est pas parce que les spammeurs ont muté et suivi les évolutions récentes qu'il ne faut pas limiter les hameçonnages qui, eux, ne les ont pas suivies.

D'autre part, les rapports reçus sont une source d'information importante sur l'usage qui est fait de notre domaine.

Et les domaines légitimes peuvent être repérés et validés. De la même manière, les domaines non légitimes seront également repérés, qu'ils signent ou non. Il sera donc facile de propager des listes de domaines nuisibles, de domaines cousins, de sources d'hameçonnages, etc. Et de se servir de ces listes, blanches ou noires, pour influer sur le traitement des messages.

Ces RFC font un usage intensif du DNS, ce qui nous expose à d'autres attaques, soulignées par les auteurs de recommandations :

- des attaques DNS en rejouant des messages comportant une signature ; si on renvoie massivement un message légitime, le nombre de requêtes DNS engendrées peut entraîner un déni de service DNS. Il est possible d'atténuer ce type d'attaque en renouvelant régulièrement les clés ;
- enfin, comme tout repose sur le DNS, sa sécurisation globale revêt une nouvelle importance pour la messagerie ; on reçoit ainsi des hameçonnages de sous-domaines extraits de domaines parfaitement légitimes, mais qui n'ont pas correctement configuré leur zone. DNSSEC retourne donc sur le devant de la scène, en tant que garant de l'intégrité du DNS sur laquelle tout repose.

Enfin, il reste un limite importante, indépendamment de tout ce que nous pourrons faire au niveau infrastructure : l'interface utilisateur.

Depuis qu'on remplace l'adresse email par le *gecos* dans l'affichage aux utilisateurs, il n'est plus forcément nécessaire d'usurper un domaine pour envoyer un hameçonnage ; si vous voyez affiché « Service clientèle banque Glzxx », cela vous donne plus facilement confiance que « reoutrdnjrezh@18gtgd.exacrrts.com ». Mais le *gecos*, c'est plus lisible, alors on laisse passer. De la même manière, quasiment aucun client ne présente d'information claire sur l'authentification serveur.

Le nombre de cas à afficher ne serait pas nécessairement très importants et sont résumés ci-dessous.

- Absence de DKIM/SPF/DMARC : classement à part possible, mais surtout avertissement : « Ce domaine ne fournit pas d'information sur son identité »
- DMARC invalide, sans ARC (ou ARC hors de la sphère de confiance) : classement en spam ; avertissement : « Possible usurpation d'identité »
- DMARC invalide mais ARC valide via un tiers de confiance : Avertissement : « Ce message ne provient pas de l'expéditeur original mais nous a été transmis par un tiers de confiance. Soyez tout de même prudent. »
- DMARC valide :
  - Situation 1 : domaine déjà connu de l'utilisateur et aucun message n'a été classé spam : message transmis sans avertissement,
  - Situation 2 : domaine pour lequel l'utilisateur n'a jamais reçu de message ou qui a déjà émis du spam : avertissement : « Ce domaine n'a pas encore fait la preuve de sa fiabilité, Traitez ce message avec méfiance ».

8. Conclusion

Nous avons donc présenté quatre RFC dont l'usage conjugué permet d'authentifier clairement la source d'un message, sans pour autant offrir de garantie sur le contenu du message. L'objectif de cet article était de rendre claires au lecteur ces recommandations trop méconnues.

Ces RFC sont progressivement adoptées par de nombreux acteurs de la messagerie. Nous sommes convaincus qu'elles peuvent représenter un excellent atout dans la lutte contre l'hameçonnage.

En revanche, nous ne commettons pas l'erreur de les considérer comme la solution unique du problème. C'est la conjonction des outils qui nous permettra de le résoudre.

On a vu que l'usage même des ces recommandations est variable d'un outil à l'autre : soit comme preuve absolue, soit comme pondération.

C'est qu'il est impossible de basculer brutalement vers une politique stricte de respect de ces recommandations. Une période de transition est nécessaire, permettant un état des lieux de la messagerie de nos établissements et un temps d'expérimentation pour rechercher la meilleurs façon de s'en servir.

Nous invitons tous les lecteurs désireux d'avancer sur le sujet à nous contacter pour qu'un travail collaboratif se mette en place afin de tordre le cou à l'hameçonnage, si pas pour toujours (faisons confiance à l'imagination des pirates) au moins pour un bout de temps.

Bibliographie

[1] Scott Kitterman, RFC 7208 : Sender Policy Framework (SPF), 2014 ; https://tools.ietf.org/html/rfc7208

[2] Dave Crocker, Tony Hansen et Murray S. Kucherawy (ed.), RFC 6376 : DomainKeys Identified Mail (DKIM) Signatures, 2011 ; https://tools.ietf.org/html/rfc6376

[3] Murray S. Kucherawy et Elizabeth Zwicky, RFC 7489 : Domain-based Message Authentication, Reporting and Conformance (DMARC), 2015 ; https://tools.ietf.org/html/rfc7489

[4] Kurt andersen, Brandon Long, Seth Blank et Murray Kucherawy, RFC 8617 : The Authenticated Received Chain (ARC) Protocol, 2019 ; https://tools.ietf.org/html/rfc8617

Abbréviations

- ADSP : Author Domain Signing Practices
- ARC : Authenticated Received Chain
- DKIM : DomainKeys Identified Mail
- DMARC : Domain-based Message Authentication, Reporting and Conformance
- DNS : Domain Name System
- GECOS : *Stricto sensu*, signifie « General Comprehensive Operating System ». Cet acronyme est employé dans certains systèmes Unix pour désigner le champ où sont stockées des informations personnelles d'un utilisateur. Par extension, le protocole SMTP emploie le même terme pour désigner la section du champ « From » contenant le nom de l'utilisateur, en texte libre.
- IETF : Internet Engineering Task Force
- IP : Internet Protocol
- MTA : Mail Transport Agent
- MX : Mail eXchanger
- RFC : Request For Comments
- SMTP : Simple Mail Transfer Protocol
- S/MIME : Secure/Multipurpose Internet Mail Extensions
- SPF : Sender Policy Framework
- SRS : Sender Rewriting Scheme

Annexes

1. Fonctionnement de SPF

1.1 Enregistrement SPF

Implémenter SPF consiste à positionner un enregistrement DNS dans lequel sont définies un ensemble de règles. Ces règles définissent les hôtes autorisés à poster des messages.

L'enregistrement est de type TXT. C'est un enregistrement sous le domaine émetteur de mail.

Exemple :

```
emetteur.tld. 14400 IN TXT "v=spf1 ip4:192.168.0.0 mx -all"
```

L'enregistrement consiste en

1. un numéro de version (« v=spf1 » dans notre exemple) ;
2. une suite de règles, qui se lisent de gauche à droite (« ip4:192.168.0.0 mx -all » dans notre exemple).

Ce sont les règles qui permettent d'évaluer la légitimité du serveur émetteur.

Une règle se compose de la façon suivante :

```
<qualificateur><mécanisme><valeurs>
```

Les qualificateurs peuvent prendre les valeurs suivantes :

- « + » : donne un résultat « pass » ; si le quantificateur n'est pas défini pour une règle, il prend par défaut la valeur « + » ;
- « - » : donne un résultat « fail » ;
- « ~ » : donne un résultat « softfail » ;
- « ? » : donne un résultat « neutral ».

Le Mécanisme peut prendre les valeurs suivantes :

- « all » : aucune contrainte ;
- « a » : correspond à l'enregistrement DNS « A » du domaine ;
- « mx » : correspond à l'enregistrement DNS « MX » du domaine ;
- « ptr » (non recommandé) : obligatoirement associé à une valeur correspondant à la spécification d'une adresse IP v4 ou v6 dont on doit vérifier qu'elle peut-être résolue en inverse sur un domaine ;
- « ip4 » : obligatoirement associé à une valeur correspondant à la spécification d'une ou plusieurs plages d'adresses IP v4 ;
- « ip6 » : obligatoirement associé à une valeur correspondant à la spécification d'une ou plusieurs plages d'adresses IP v6 ;

– « include » : obligatoirement associé à une valeur correspondant à un nom de domaine dont on va évaluer l'enregistrement SPF.

– « exists » : obligatoirement associé à une valeur correspondant à une spécification formelle d'un enregistrement.

La valeur dépend du mécanisme employé. Pour « ip4 », par exemple, la valeur est une spécification d'adresses IP v4. Certains mécanismes, comme « all » ou « a » ne contiennent pas de valeur.

Reprenons notre exemple. L'ensemble des règles est :

```
ip4:192.168.0.0 mx -all
```

Nous avons donc trois règles :

– « ip4 » :
  – Quantificateur : « + » (valeur par défaut),
  – Mécanisme : « ip4 »,
  – Valeur : « 192.168.0.0 » ;
– « mx » :
  – Quantificateur : « + » (valeur par défaut),
  – Mécanisme : « a »,
  – Valeur : aucune ;
– « -all » :
  – Quantificateur : « - »,
  – Mécanisme : « all »,
  – Valeur : aucune ;

Dans notre exemple, si un mail arrive soit de l'hôte ayant l'IP 192.168.0.0, soit d'un hôte défini dans l'enregistrement « MX », l'évaluation de SPF doit donc donner un résultat « pass ». Dans tous les autres cas (mécanisme « all »), elle produit un résultat « fail ».

Il est possible de déporter tout ou partie de la définition des règles de deux manières. Soit en utilisant le mécanisme « include », soit en utilisant le mécanisme « redirect ». Dans les deux cas, on évalue l'enregistrement SPF définit pour le domaine contenu par la valeur du mécanisme. Que l'on écrive « redirect:autre.emetteur.tld » ou « include:autre.emetteur.tld », on évaluera de toutes façons le SPF de « autre.emetteur.tld ».

La différence se situe dans ce qu'on fait des résultats.

Dans le cas d'un include, si le domaine émetteur de mail vérifie une des règles de l'enregistrement SPF inclus, l'évaluation renvoie « match » et on applique alors le résultat du quantificateur placé devant « include ». Dans le cas d'un match, si la règle était « +include », l'évaluation SPF renvoie finalement « pass ». Mais dans le même cas, si la règle était « -include », l'évaluation renvoie « fail ».

Dans le cas d'un redirect, on évalue les règles SPF du domaine vers lequel a lieu la redirection. Deux nuances majeures sont introduites ici :

1. Le domaine employé n'est pas le domaine d'orgine mais le domaine de redirection. Si l'enregistrement SPF du domaine « a.tld » contient « redirect:b.tld », alors on évalue l'enregistrement de « b.tld » avec comme valeur de domaine, « b.tld ».
2. Le résultat de l'évaluation de la redirection devient le résultat de l'évaluation courante. Il n'y a pas de quantificateur devant le mot « redirect » qui pourrait transformer un « pass » en « fail », comme pour un include.

Notons enfin que l'on peut ajouter, dans l'enregistrement, la chaîne « exp: » suivie d'une explication en texte permettant de donner des indications sur les règles SPF, à destination d'un administrateur de messagerie désireux de comprendre ce que l'on a fait.

1.2 Évaluation

À la réception d'un mail, un serveur supportant SPF va évaluer les données du mail en fonction de l'enregistrement du domaine émetteur.

Un mot important à propos du domaine. Dans le cadre de SPF, l'adresse IP du MTA émetteur, le domaine d'expédition extrait du « MAIL FROM » de l'enveloppe et éventuellement le domaine indiqué lors de la commande « EHLO / HELO » de la session SMTP, sont recueillis. C'est une distinction importante car il s'agit du seul des quatre protocoles présentés dans l'article qui s'appuie sur l'enveloppe du mail. Les trois autres s'appuient sur les entêtes.

Après une validation de la syntaxe de l'enregistrement, les mécanismes sont évalués de gauche à droite. Si les données du message correspondent, on applique le qualificateur. Sinon, on passe au mécanisme suivant, et ainsi de suite. L'évaluation s'arrête dès que l'on obtient une correspondance ou qu'il ne reste plus de mécanismes à évaluer. Dans ce dernier cas, en renvoie la valeur de qualificateur par défaut : « neutral ».

L'évaluation peut renvoyer les valeurs suivantes :

- « none » : quand il n'a pas été possible d'obtenir les informations nécessaires à l'évaluation. Par exemple, en l'absence d'enregistrement SPF ;
- « neutral » : si l'évaluation d'un règle permet de renvoyer le qualificateur « ? » ou que toutes les règles ont été évaluées sans obtenir de correspondance ;
- « pass » : si l'évaluation d'un règle permet de renvoyer le qualificateur « + » ;
- « fail » : si l'évaluation d'un règle permet de renvoyer le qualificateur « - » ;
- « softfail » : si l'évaluation d'un règle permet de renvoyer le qualificateur « ~ » ;
- « temperror » : erreur temporaire ; peut arriver dans le cas d'un indisponibilité temporaire de DNS ;
- « permerror » : erreur permanente, par exemple une erreur de syntaxe de l'enregistrement DNS.

Le résultat de l'évaluation est stocké soit dans un entête « Received-SPF », soit dans un entête « Authentication-Results », soit les deux.

Par exemple, on peut trouver ces entêtes après un SPF évalué à « pass » :

```
Received-SPF: pass (destination.tld: domain of
utilisateur@origine.tld designates 192.168.0.0 as permitted
sender) client-ip=192.168.0.0;
Authentication-Results: mx.destination.tld;
       spf=pass (destination.tld: domain of
utilisateur@origine.tld designates 192.168.0.0 as permitted
sender) smtp.mailfrom=utilisateur@origine.tld;
```

À partir de là, c'est au serveur destinataire d'appliquer un traitement adéquat au message en fonction de la valeur positionnée dans les entêtes SPF.

2. Fonctionnement de DKIM

DKIM, c'est un enregistrement DNS et une signature en fin de message.

Un enregistrement DKIM a l'aspect suivant :

```
dkim._domainkey.emetteur.tld. 1779 IN TXT "v=DKIM1;
k=rsa;p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDI6mZ3PHIbGNtNruAsb
F/wNcqyNiDT0CReSZQc9yVp6i24MxfLlOg+
+RhVJ0r8V0bpBLr34yVOGbsjK0VLbN9Xjusrs9qRlvOoAXaOTRZSiIFBXgbp7AY0n
wmOoFYkjmf/2FUq5szjl8rom8bXM7TlAcLmvxWOi0e+jn5T0z888QIDAQAB"
```

L'entrée DNS suit toujours le format suivant :

```
<sélecteur>._domainkey.<domaine>
```

- le sélecteur est une chaîne arbitraire,
- « _domainkey » est fixe et sert à délimiter le sélecteur et le domaine,
- Le domaine est le domaine mail de l'émetteur.

Concrètement, c'est un enregistrement de type TXT pouvant contenir les paires « clé=valeur » suivantes :

- « v=valeur » : la version de DKIM employée. « DKIM1 » est la seule valeur valide ;
- « h=<valeur> » : algorithme de hachage employé pour hacher les données du message avant de les signer. Deux valeurs possibles : « sha1 » ou « sha256 » ;
- « k=<valeur> » : algorithme de signature des données hachées. Une seule valeur possible : « rsa »
- « n=<valeur> » : des notes à destination d'un utilisateur humain
- « p=<valeur> » : la clé publique, utilisée pour valider la signature d'un message ; Il est possible et souvent employé dans le cas de mutualisation, d'utiliser une paire spécifique, par expéditeur, par domaine.. selon le niveau de confiance ou de rotation des clef qui est prévu… Chaque mail signé embarque un sélecteur qui permet de choisir la bonne clef publique pour faire la vérification.
- « s=<valeur> » : le sélecteur,
- « t=<valeur> » : Des drapeaux séparés par « : ». Si cette clé contient le drapeau « y », le traitement des messages ne changera pas, mais on pourra recevoir des retours de la part des destinataires sur

l'état de signature constaté. Dans la pratique, cela ne se produit jamais. Si elle contient le drapeau « s », aucune signature pour un sous-domaine du domaine utilisé dans le DNS ne sera considérée valide. Si l'on n'utilise pas de sous-domaines de messagerie, il est recommandé de positionner ce drapeau.

La signature DKIM est placée dans l'entête « DKIM-Signature ». Elle a la structure suivante :

```
DKIM-Signature: v=1; a=rsa-sha256; c=simple/simple;
d=emetteur.tld; s=2018; t=1566993905;
bh=9rmOIcS+2UrInN2TCX/LQY4213N8rhXcXISALtMT+c0=;
h=To:From:Subject:Date:From; b=kzgg9JLzTs7257g...
```

La signature peut contenir la liste de paires « clé=valeur » suivante :

- v=<valeur> : la version de DKIM employée ; Seule la valeur « 1 » est acceptée ;
- a=<signature>-<hachage> : les algorithmes employés pour générer la signature ; <signature> est l'algorithme de calcul de la signature (une seule valeur possible : « rsa ») et <hachage> est l'algorithme de hachage (« sha1 » ou « sha256 ») ;
- b=<valeur> : la signature du message, en base 64 ;
- bh=<valeur> : le corps du message haché ;
- c=<algorithme_entêtes>/<algorithme_corps> : les algorithmes de canonicalisation employés respectivement pour les entêtes et le corps du message ; Cette canonicalisation permet d'éviter des confusions, notamment dans la gestion des espaces lors du calcul de la signature ;
- d=<valeur> : le domaine signataire. Combiné à la clé « s= », il permet de reconstruire l'enregistrement DNS contenant la clé publique nécessaire à la vérification de la signature ;
- h=<valeur> : la liste des entêtes du message faisant partie de la signature, séparés par des « : » ;
- i=<valeur> : dans le cas où le message vient d'un sous-domaine de celui indiqué dans « d= », celui-ci doit figurer ici. Notons que si l'enregistrement DNS a positionné le drapeau « s », aucun sous-domaine ne peut y figurer ;
- l=<valeur> : la longueur, en octets, du corps du message canonicalisé ;
- q=<valeur> : la méthode à employer pour récupérer la clé publique. Actuellement, seule la valeur « dns /txt » est acceptée. Cette clé est présente pou permettre l'éventuelle introduction d'autres méthodes dans l'avenir ;
- s=<valeur> : le sélecteur, utilisé conjointement à la clé « d= » pour récupérer la clé publique ;
- t=<valeur> : le timestamp de la date de signature ;
- x=<valeur> : le timestamp de la date d'expiration de la signature. Au-delà de cette date, une signature doit être considérée invalide ; Par défaut, les signatures n'expirent pas ;
- z=<valeur> : une liste, séparée par des barres verticales (« | »), des entêtes présents lors de la signature, ainsi que leur valeur. Détail cocasse : les entêtes n'ont pas nécessairement à être réellement présents au moment de la signature. On peut parfaitement y mettre une chaîne telle que « Entete: ». Sans aucune valeur en dehors du nom de l'entête. De cette manière, si « Entete: » apparaissait dans le message final, on saurait qu'il a été ajouté en cours de route. Ceci n'a pas d'influence sur l'évaluation de la signature, seulement pour établir des diagnostics.

Un serveur recevant un mail signé DKIM va donc trouver cet entête, récupérer la clé publique et valider la signature.

Cette validation peut avoir trois résultats :

- « success » : la signature a pu être vérifiée et elle est valide ;
- « permfail » : la signature est invalide ou l'enregitrement DNS n'existe pas ;
- « tempfail » : la signature n'a pu être vérifiée, par exemple pour des questions d'indisponibilité DNS.

Une fois la signature validée, c'est au destinataire de décider quoi faire. Dans bon nombre de cas, le résultat de la validation de la signature est stocké dans le message via un entête « Authentication-Results », à l'instar de ce qui peut se faire pour SPF.

3. Fonctionnement de DMARC

Pour définir une politique DMARC, il faut et il suffit de positionner un enregistrement DNS de type TXT.

3.1 L'enregistrement DNS DMARC

Par exemple :

```
_dmarc.<emetteur.tld> 1800 IN      TXT      "v=DMARC1; p=reject;
rua=mailto:local@emetteur.tld; ruf=mailto:local@emetteur.tld;
fo=1; pct=100"
```

Le nom de l'enregistrement a toujours la structure suivante :

```
_dmarc.<domaine.tld>
```

- « _dmarc » est fixe et sert à délimiter le sélecteur et le domaine,
- <domaine.tld> est le domaine mail de l'émetteur.

L'enregistrement lui-même peut contenir les paires « clé=valeur » suivantes :

- v=<valeur> : la version de DMARC employée ; une seule valeur valide : « DMARC1 ».
- p=<valeur> : la politique de sécurité. Trois valeurs possible : « reject », « quarantine » ou « none » ;
- sp=<valeur> : la politique de sécurité pour les sous-domaines. Trois valeurs possible : « reject », « quarantine » ou « none » ;
- pct=<valeur> : le pourcentage de mails auxquels appliquer la politique DMARC ;
- adkim=<valeur> : type d'alignement (r pour « relaxed » ou s pour « strict ») à respecter du point de vue de DKIM ;
- aspf=<valeur> : type d'alignement (r pour « relaxed » ou s pour « strict ») à respecter du point de vue de SPF;
- fo=<valeur> : série d'options pour spécifier pour quelles combinaisons d'erreurs d'alignement un rapport sera généré ;

– rf=<valeur> : format à employer pour envoyer les rapports d'erreurs ; une seule valeur possible : « atrf » ;

– ri=<valeur> : intervalle demandé, en secondes, entre deux envois de rapports agrégés ; 86400 par défaut ; la plupart des implémentations utilisent cette valeur quoi qu'on positionne ;

– rua=<valeur> : URI à employer pour transmettre les rapports agrégés ; les URI de type « mailto : » sont partout supportées ; aucune garantie de support sur un autre protocole.

– ruf=<valeur> : URI à employer pour transmettre les rapports d'erreurs ; les URI de type « mailto : » sont partout supportées ; aucune garantie de support sur un autre protocole.

3.2 Authentification et alignement

Quand un serveur supportant DMARC reçoit un mail, il recherche un enregistrement DNS DMARC pour le domaine d'expédition. Le domaine, ici, est choisi en extrayant la partie domaine de l'adresse email trouvée dans le champ « From » du message.

Si SPF ou DKIM (aucune obligation que les deux soient simultanément valides) est évalué à « pass », alors DMARC est évalué à « pass ».

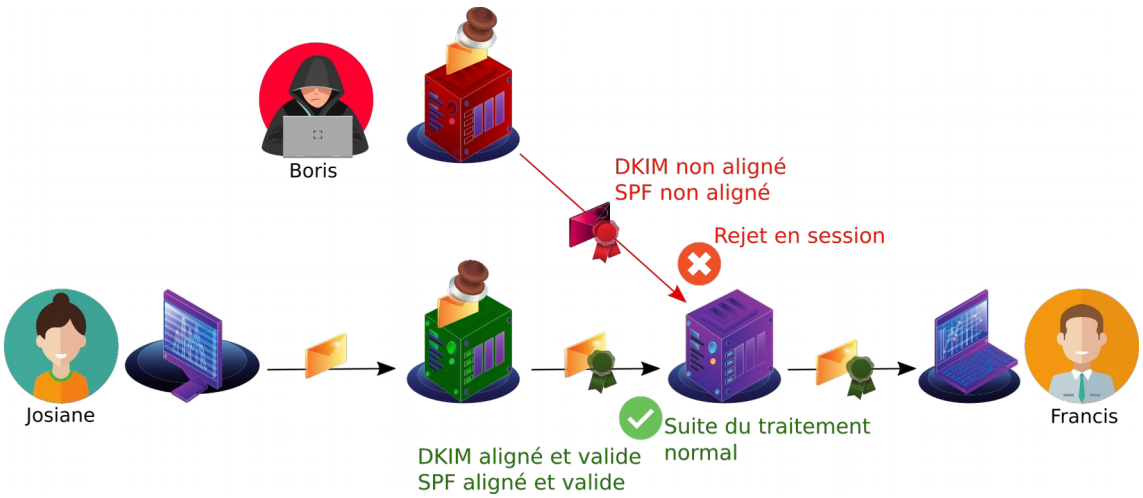
Mais attention : ça ne marche que s'il y a alignement.

Le principe de l'alignement est le suivant : les vérifications SPF et DKIM sont faites en se fondant sur le domaine indiqué dans le champ « From ». Toute signature DKIM émise depuis un autre domaine, ou tout serveur émetteur absent du SPF de ce domaine seront ignorés.

L'alignement peut être plus ou moins strict. Le type d'alignement est précisé dans les clés « aspf » et « adkim » de l'enregistrement DNS. Ces deux clés peuvent avoir les valeurs suivantes :

– « r » (pour « relaxed », valeur par défaut) : les sous-domaines sont considérés alignés avec le domaine principal ;

– « s » (pour « strict ») : il n'y a alignement que si la partie locale du champs « From » et le domaine présenté (soit dans le champs « d= » de la signature DKIM, soit dans l'enregistrement DNS SPF) sont strictement identiques. En bref : pas de sous-domaine.



DMARC définit une nouvelle valeur pour le champ « Authentication-Results », la valeur « dmarc ». Celle-ci vaut « pass » si doit DKIM, soit SPF, soit les deux valent « pass ».

4. Fonctionnement d'ARC

Techniquement, sceller le message se fait en apposant trois entêtes qui constituent ce que les auteurs de la RFC appellent un « ARC set » et que nous traduirons par « triplet ARC ».

Les trois entêtes d'un triplet ARC sont :

- ARC-Authentication-Results : spécifie dans quel état d'authentification (spf, dkim, dmarc et arc) le message a été reçu ; il est quasiment identique à un entête « Authentication-Results ».
- ARC-Message-Signature : Signature du message à l'expédition. Cette signature est quasiment identique à une signature DKIM ;
- ARC-Seal: Signature de la chaîne ARC. Cette signature est quasiment identique à une signature DKIM, à ceci près qu'elle n'intègre que les seuls entêtes ARC et fait comme si le corps du message était vide.

Vous remarquerez que nous avons écrit « quasiment identique » dans la description des trois entêtes. En-dehors des modifications précisées dans la liste ci-dessus, une modification apparaît das les entêtes ARC, par rapport à leur modèle : une paire « clé=valeur » indiquant l'indice du triplet : « i= ».

Cette clé existe dans les signatures DKIM mais elle n'a pas le même rôle pour ARC : À chaque fois qu'un participant à la chaîne ajoute ses entêtes, il les numérote tous avec la même valeur, dans « i= ».Le premier participant à la chaîne donne la valeur « i=1 » aux trois entêtes de son triplet, le second « i=2 » et ainsi de suite.

L'entête « ARC-Message-Signature » peut donc contenir la liste de paires « clé=valeur » suivante :

- toutes les paires « clés=valeur » définies par DKIM sauf « i= » qui prend un autre sens, précisé ci-dessous, et « v= », proscrit ;
- i=<valeur> : l'indice du triplet auquel il appartient.

L'entête « ARC-Seal » peut, pour sa part contenir la liste de paires « clé=valeur » suivante :

- les paires « clés=valeur » définies par DKIM suivantes (voir annexe 2 pour leur définition) :
  - « a= »
  - « b= »
  - « d= »
  - « s= »
  - « t= »
- i=<valeur> : l'indice du triplet auquel il appartient.
- cv=<valeur> : indique quel est l'état de validation de la chaîne d'authentification ARC au moment où ce triplet a été généré. Elle peut prendre trois valeurs :
  - « pass », si la chaîne es valide,
  - « fail », si elle est invalide,

– « none » s'il n'y a pas encore de chaîne, comme dans le cas d'un triplet positionné par l'expéditeur du message, ou si le message est reçu sans chaîne ARC.

L'entête « ARC-Authentication-Results » peut, pour sa part contenir la liste de paires « clé=valeur » suivante :

– toutes les paires « clés=valeur » autorisées dans un entête « Authentication-Results » ;

– i=<valeur> : l'indice du triplet auquel il appartient.

Par exemple, dans les entêtes ci-dessous se trouvent deux triplets ARC.

```
ARC-Seal: i=2; a=rsa-sha256; t=1566306919;
cv=pass;d=destinataire.tld; s=arc-20160816;b=Nsz[…]

ARC-Message-Signature: i=2; a=rsa-sha256;
c=relaxed/relaxed;d=destinataire.tld; s=arc-20160816;h=mime-
version:to:reply-to:from:subject:date:message-id:dkim-
signature;bh=4ve8[...]=;b=l2c[…]

ARC-Authentication-Results: i=2; mx.destinataire.tld;dkim=pass
header.i=@emetteur.tld header.s=dkim header.b=jBxBdFaw;arc=pass
(i=1);spf=pass (destinataire.tld: domain of contact@emetteur.tld
designates 192.168.0.0 as permitted sender)

[…]

ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed;
d=emetteur.tld; s=dkim; t=1566306918; h=from:from:reply-to:reply-
to:subject:subject:date:date:message-id:message-id:to:to:cc:mime-
version:mime-version:content-type:content-
type;bh=4ve[...];b=k1V[…]

ARC-Seal: i=1; s=dkim; d=emetteur.tld; t=1566306918;a=rsa-sha256;
cv=none;b=cfI[…]

ARC-Authentication-Results: i=1; emetteur.tld; none
```

Le premier dans le sens de lecture (portant l'indice « i=2 ») a été apposé par le serveur destinataire. Le second triplet dans le sens de lecture (portant l'indice « i=1 ») a été apposé par le serveur émetteur.

Le triplet du destinataire (indice « i=2 ») a été apposé après vérification de l'authentification serveur ; on constate en lisant l'entête « ARC-Authentication-Results » qu'en effet, SPF et DKIM donnent un résultat « pass ». Dans cet entête se trouve également une troisième méthode d'authentification introduite par la RFC : « arc ».

Le triplet de l'émetteur (indice « i=1 ») a été apposé juste avant l'expédition du message et on peut remarquer que l'entête « ARC-Authentication-Results » contient la valeur « none ». En effet, étant à l'origine du message, aucun résultat d'authentification ne pré-existait lors de la création du triplet.

L'entête « ARC-Seal » d'indice 1 possède la valeur « none » pour la clé « cv= » car c'est lui qui a créé la chaîne d'authentification. L'entête « ARC-Seal » d'indice 2 possède la valeur « pass » pour la clé « cv= » car la chaîne était valide quand il l'a reçue.

4.1 Évaluation

Tout d'abord, un vérificateur (nom donné à un serveur évaluant un message avec un chaîne ARC) vérifie l'intégrité de la chaîne d'authentification :

- présence des triplets pour chaque étape ;
- concordance de la numérotation des indices ;
- présence d'un clé « cv=pass » dans tous les entêtes « ARC-Seal ».

Ensuite, il vérifie la validité de la signature de l'entête « ARC-Message-Signature » de plus grand indice.

Enfin, il vérifie la validité la signature de chaque entête « ARC-Seal » en partant du plus grand indice.

Si une seule de ces étapes échoue, le statut de validité de la chaîne devient « fail ». Sinon, il devient « pass ».

Dès lors que la chaîne d'authentification a le statut « pass », le vérificateur sait que les informations contenues par la chaîne sont fiables. En particulier, il peut avoir confiance dans le fait que la valeur des différents entêtes « ARC-Authentication-Results » ont bien été apposés par un agent de leurs domaines respectifs.

5. Exemple de message traversant l'infrastructure de test

Nous avons reproduit plus bas les entêtes d'un message qui a traversé cette infrastructure. Nous avons découpé ces entêtes et leur avons adjoint un numéro correspondant à l'étape de traitement. Les étapes de traitement sont détaillées ci-dessous. Notez que, la plpuart du temps, les entêtes sont ajouté en haut de ceux pré-existants. Les étapes apparaissent donc en sens inverse du sens de lecture.

Voici les étapes :

1 l'expéditeur fabrique son message et le signe avec DKIM ;

2 le message est reçu chez RENATER :

2.1 SPF échoue (car c'est l'anti-spam entrant qui nous transmet le courriel) ;

2.2 mais la signature DKIM est valide et permet la validation DMARC de l'expéditeur ;

3 le message est transmis au serveur de listes qui prépare sa distribution ;

4 les mails générés sont transmis aux MTAOut ;

5 les mails sortants sont signés DKIM et scellés ARC (optionnel à cette étape puisque Authentication-Results est vide) ;

6 les IP des MTAOut sont renseignés dans l'enregistrement SPF pour le domaine du serveur de listes ;

7 la signature DKIM est valide et alignée avec le « From » du message, les domaines dans le « MAIL FROM » de l'enveloppe et le champ « From » des entêtes sont alignés, DMARC est valide ;

8 Google valide et scelle ARC.

Exemple sur le message suivant :

```
Delivered-To: destinataire@gmail.com
Received: by 2002:a19:df41:0:0:0:0:0 with SMTP id q1csp2295031lfj; Thu, 12 Sep 2019 07:48:11 -0700 (PDT)
X-Google-Smtp-Source:
APXvYqy0VXjBKKEQrywzeBF+TxblCmPasnj7G7gFNURSSalO3MHPrV3/TfHxrRIcpNaKwRGI7ekU
X-Received: by 2002:adf:dec2:: with SMTP id i2mr19329763wrn.92.1568299691250;
        Thu, 12 Sep 2019 07:48:11 -0700 (PDT)
```

8

```
ARC-Seal: i=2; a=rsa-sha256; t=1568299691; cv=pass;
        d=google.com; s=arc-20160816;
        b=bAZPXCDKwzqxiNyYsYzoPPfB14patoeRM51zZVmIpwgev+cp5K/DUGmCoeAv27zcpK
         FMrgd8cdPGDC1/1xT0GqIjzs+0dkoQ1ZredMtXm0TB4pmcD8TilfuEDDF62WtM9pK0kR
         FquNBVPVfSV3OoAGWN5uc083PEffPeisHClCq3xteQix5nCfpw9ftfN7pivoN1hffg46
         roaqeJNfcnc9EpWvE/1YumHO4CVSG5RgwbuVU4PpsBmMj4oP24Fl5S1fy/g974EvkF/5
         P/rDUwUCsp6E3tKOAYwpUYCukqSBGOC8VhA6dCWu7lK1hvQdeYpZ5XZ1RGBckL3LSrmY
         wl6A==
ARC-Message-Signature: i=2; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816;
        h=archived-at:list-archive:list-owner:list-post:list-unsubscribe
         :list-subscribe:list-help:list-id:sender:precedence:precedence
         :errors-to:reply-to:subject:content-language
         :content-transfer-encoding:in-reply-to:mime-version:user-agent:date
         :message-id:from:references:to:dkim-signature:dkim-filter
         :dkim-signature:dkim-filter:dkim-signature;
        bh=QJxbwdRTg9fsOZBfha2OCdzYLQu5i5slwHrwHSwByN4=;
        b=uI3AhcI7cyXrLKVOMtus8cmZgWd+XxTKaDUF9XjZoiubeXbamdB3Jn7Dhe2fQBieS8
         FVbIMh64u/9G2Uenv4IjdWOmbjV21eKBj+myHN6ZPPaoLFXTGNkva9l0OJYCeXGAt6tR
         YV2iPERf43ZBKkHrhENwXnvKtG/rumBGIZquUVRpSZ1nme3wJSTwpBTO7L+lBnrSZPOG
         uwYvh1kvBTaLAPrhWNmo2W89BxBmLIb2wWTSL7xNZzSkOI2g3op+bTNcEIOhhicMBWSG
         P3n/2DVGLVaAK55wkkRSXLmXbu3xPZYi9VewZgUIozUVTjczyogiR2NHYcVRLlK/9SzG
         0Pig==
ARC-Authentication-Results: i=2; mx.google.com;
       dkim=pass header.i=@groupware2.renater.fr header.s=groupware2 header.b=iikMHQYM;
       dkim=pass header.i=@domaine-expediteur.fr header.s=2018 header.b=H1IHTF9z;
       arc=pass (i=1);
        spf=pass (google.com: domain of testdmagp2-owner@groupware2.renater.fr designates 194.57.4.214 as permitted sender) smtp.mailfrom=testdmagp2-owner@groupware2.renater.fr;
       dmarc=pass (p=REJECT sp=REJECT dis=NONE) header.from=domaine-expediteur.fr
```

```
Return-Path: <testdmagp2-owner@groupware2.renater.fr>
Received: from ix1-bv-c7-UniversalistesMTAOut-01.renater.fr (ix1-bv-c7-universalistesmtaout-01.renater.fr. [194.57.4.214])
        by mx.google.com with ESMTPS id q12si133654wmc.196.2019.09.12.07.48.10
        (version=TLS1_2 cipher=ECDHE-RSA-AES128-GCM-SHA256 bits=128/128);
        Thu, 12 Sep 2019 07:48:11 -0700 (PDT)
```

6

```
Received-SPF: pass (google.com: domain of testdmagp2-owner@groupware2.renater.fr designates 194.57.4.214 as permitted sender) client-ip=194.57.4.214;
```

7

```
Authentication-Results: mx.google.com;
       dkim=pass header.i=@groupware2.renater.fr header.s=groupware2 header.b=iikMHQYM;
       dkim=pass header.i=@domaine-expediteur.fr header.s=2018 header.b=H1IHTF9z;
       arc=pass (i=1);
```

spf=pass (google.com: domain of testdmagp2-owner@groupware2.renater.fr designates 194.57.4.214 as permitted sender) smtp.mailfrom=testdmagp2-owner@groupware2.renater.fr;
dmarc=pass (p=REJECT sp=REJECT dis=NONE) header.from=domaine-expediteur.fr

4
Received: from ix1-bv-u16-Universalistessympa-15 (unknown [194.57.4.247])
by ix1-bv-c7-UniversalistesMTAOut-01.renater.fr (Postfix) with ESMTP id 46ThTQ5hFrz2D6W;
Thu, 12 Sep 2019 16:48:10 +0200 (CEST)

5
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=groupware2.renater.fr;
s=groupware2; t=1568299690;
h=from:from:sender:sender:reply-to:reply-to:subject:subject:date:date:
message-id:message-id:to:to:cc:mime-version:mime-version:
content-type:content-type:
content-transfer-encoding:content-transfer-encoding:
in-reply-to:in-reply-to:references:references:list-id:list-help:
list-owner:list-unsubscribe:list-subscribe:list-post;
bh=QJxbwdRTg9fsOZBfha2OCdzYLQu5i5slwHrwHSwByN4=;
b=iikMHQYMcljSWeB/ET8rAal/2VjTtL8fKrCsAgCleNkTjaUzJP2rRMs7+bO3Uze3TnKM+z
oTtmoaFK8+SKXXtjRgS/oC8dn3yWJuzhUDWVMSOrhbSS/wUDISrEHsn9qLYub3zFDrrZyD
NcRGn2O9uessvkgmLQm14ltWBwnIUIlOJ5H8o8l77y8Be1gQ7GyfJmHhY8vdz+gbIUs00O
S+4/L3e67RdKPhMQadjWRq/BEeua47NtiD/FxXIq0RxZo01Kv77yMueUFr5oDxBTB46xuz
tmcft/e4yDT5+oMM4g8qy2xioCjExkucoZlUg49S5oTNkJ39kbrOyN2ATpLRFw==

Received: by ix1-bv-u16-Universalistessympa-15 (Postfix, from userid 50002)
id AF4A71F701; Thu, 12 Sep 2019 16:47:38 +0200 (CEST)

3
Received: from ix1-bv-c7-UniversalistesMTAIn-02.renater.fr (ix1-bv-c7-universalistesmtain-02.renater.fr [194.57.4.215])
by ix1-bv-u16-Universalistessympa-15 (Postfix) with ESMTPS id 78F781F6C1
for <testdmagp2@groupware2.renater.fr>; Thu, 12 Sep 2019 16:47:35 +0200 (CEST)

Received: from mxb2-1.relay.renater.fr (unknown [194.57.4.247])
by ix1-bv-c7-UniversalistesMTAIn-02.renater.fr (Postfix) with ESMTP id 46ThTM320Gz1Fhbb
for <testdmagp2@groupware2.renater.fr>; Thu, 12 Sep 2019 16:48:07 +0200 (CEST)

2.2
Authentication-Results: ix1-bv-c7-UniversalistesMTAIn-02.renater.fr;
arc=none (no signatures found);
dkim=pass (2048-bit rsa key sha256) header.d=domaine-expediteur.fr header.i=@domaine-expediteur.fr header.b=H1lHTF9z header.a=rsa-sha256 header.s=2018;
dmarc=pass policy.published-domain-policy=reject policy.applied-disposition=none policy.evaluated-disposition=none (p=reject,d=none,d.eval=none) policy.policy-from=p header.from=domaine-expediteur.fr;
spf=fail smtp.mailfrom=david@domaine-expediteur.fr smtp.helo=mxb2-1.relay.renater.fr;
x-aligned-from=pass (Address match);
x-google-dkim=none (no signatures found);
x-return-mx=pass header.domain=domaine-expediteur.fr policy.is_org=yes (MX Records found: domaine-expediteur.fr);
x-return-mx=pass smtp.domain=domaine-expediteur.fr policy.is_org=yes (MX Records found: domaine-expediteur.fr)

2.1
Received-SPF: fail
(domaine-expediteur.fr: Sender is not authorized by default to use 'david@domaine-expediteur.fr' in 'mfrom' identity (mechanism '-all' matched))
receiver=ix1-bv-c7-UniversalistesMTAIn-02.renater.fr;
identity=mailfrom;
envelope-from="david@domaine-expediteur.fr";
helo=mxb2-1.relay.renater.fr;

```
  client-ip=194.57.4.247
Received: from domaine-expediteur.fr (domaine-expediteur.fr [78.241.223.87])
        (using TLSv1.2 with cipher AECDH-AES256-SHA (256/256 bits))
        (No client certificate requested)
        by mxb2-1.relay.renater.fr (asm) with ESMTPS id D8D016076B
        for <testdmagp2@groupware2.renater.fr>; Thu, 12 Sep 2019 16:48:06 +0200 (CEST)
Received: from localhost (localhost [127.0.0.1])
        by domaine-expediteur.fr (Postfix) with ESMTP id BF441832A2
        for <testdmagp2@groupware2.renater.fr>; Thu, 12 Sep 2019 16:48:04 +0200 (CEST)
1 DKIM-Filter: OpenDKIM Filter v2.10.3 domaine-expediteur.fr BF441832A2
DKIM-Signature: v=1; a=rsa-sha256; c=simple/simple; d=domaine-expediteur.fr; s=2018;
        t=1568299684; bh=wof37tK0qBzZniJGkKE97+ZV6Dp2GBgfCml6wLLRIH8=;
        h=Subject:To:References:From:Date:In-Reply-To:From;
        b=aG4JK8VPVgYiK6GvuHS8qAcGs+MsfWmFKpStdRmUXxZko6CdP/Jdk9hcQToi9siP6
         ZXzuoxNNKzt4Z2R2nyhMTZwjIS6r73zdCMb+zcz+mikLJjZSsVCN++9Yc+Z4DrFF6Q
         Bsg9KG27AK8uZzyzAX/6rOFvvJSv+g86lqjVGU1/v1X1G9aqnYWBBxRICoqC/jmcAs
         puGHhcsaJ6ANKlUbuQ80VxsJaglphvIsRABi9XEcy8F728I/eLl1qw7yZatRIDIex/
         11eFHc/PLipuH7PPHDocS8YxRZgbq2ea1tkbGAFihG9CtdsK+My62xGx33Gofr23KM
         lDegPFmvFqWsA==
X-Virus-Scanned: Debian amavisd-new at domaine-expediteur.fr
Received: from domaine-expediteur.fr ([127.0.0.1])
        by localhost (gaia.domaine-expediteur.fr [127.0.0.1]) (amavisd-new, port 10024)
        with ESMTP id q0X6QUVJHUVI for <testdmagp2@groupware2.renater.fr>;
        Thu, 12 Sep 2019 16:48:01 +0200 (CEST)
Received: from [195.220.94.98] (mac-dv.cru.fr [195.220.94.98])
        by domaine-expediteur.fr (Postfix) with ESMTPSA id 53B8081318
        for <testdmagp2@groupware2.renater.fr>; Thu, 12 Sep 2019 16:48:01 +0200 (CEST)
To: testdmagp2@groupware2.renater.fr
From: David Verdin <david@domaine-expediteur.fr>
Message-ID: <2f0b6482-c618-768e-54bd-7f8429f426f8@domaine-expediteur.fr>
Date: Thu, 12 Sep 2019 16:46:53 +0200
MIME-Version: 1.0
Content-Type: text/plain; charset=utf-8; format=flowed
X-Authentication-Milter: Header added by Authentication Milter
Subject: Re: [testdmagp2] test rspamd allow mismatch
Reply-To: testdmagp2@groupware2.renater.fr
Sender: testdmagp2-request@groupware2.renater.fr
List-Id: <testdmagp2.groupware2.renater.fr>
List-Help: <mailto:sympa@groupware2.renater.fr?subject=help>
List-Subscribe: <mailto:sympa@groupware2.renater.fr?subject=subscribe%20testdmagp2>
List-Unsubscribe: <mailto:sympa@groupware2.renater.fr?subject=unsubscribe%20testdmagp2>
List-Post: <mailto:testdmagp2@groupware2.renater.fr>
List-Owner: <mailto:testdmagp2-request@groupware2.renater.fr>
List-Archive: <https://groupware2.renater.fr/sympa/arc/testdmagp2>
5 ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed;
        d=groupware2.renater.fr; s=groupware2; t=1568299690;
        h=from:from:sender:sender:reply-to:reply-to:subject:subject:date:date:
         message-id:message-id:to:to:cc:mime-version:mime-version:
         content-type:content-type:
         content-transfer-encoding:content-transfer-encoding:
         in-reply-to:in-reply-to:references:references:list-id:list-help:
```

```
        list-owner:list-unsubscribe:list-subscribe:list-post:dkim-signature;
        bh=QJxbwdRTg9fsOZBfha2OCdzYLQu5i5slwHrwHSwByN4=;
        b=pXIKAtXpfzcym8jyqV4A2MxqI336hwy1RZyadmzYs5FdUOiTsTplRK+yI7/BMD6Z5KXs1Q
        XvnPu1Dxm2x+kHJLTy5Ehs7qjo1SzG35U0kxbZ0nnaB+FBZrQzblbA6sikEW+AzZJxzd5w
        SnsAoVpO187+dk5H+vd9kgMbuetFUEPGFCne5IqRysf8eRw6eRCJLwPMeCMnIcEoVVqnx0
        FlEMANxM3wU2/6mHTCPiXxRFo5EwTZGMhbvQLUNp91ctNfr17aOLVdwX1+ivoh3JY9KLir
        B4Fzdw2dECVk8CSTe10lJ6lFz8+Rm3WBC5S9NmB2VaTsqEV5GklVhF3TiREpqw==
ARC-Seal: i=1; s=groupware2; d=groupware2.renater.fr; t=1568299690;
        a=rsa-sha256; cv=none;
        b=s49cBQcC8D+Ugypjz7NSF6ijleI9XFT6EuhMZGLf9tgcCSOZBSb++X/zoBoXlX6IotNqeK
        9DToMNDIbqiURjxbYTbNtOpmfJGFsGN1992HKzS5Z62qERQXQWI8i105mqrvfCVEL+r7fx
        +9n3Ep84jPWSWP2zqDqd8H52sNNSLj6H7nT56iQy9LLwB80Qdcg/mfcuj+xN3BSObm/Fs0
        76kAhzTSClHoYocYvWr0DkTbnFXBy3FVretGoRLvDVDUM2ekd/7Z7AVlhs2ElueU3XrmoY
        431IbXIJ9mHoyqGz9Tyj72RXo20HIl8fwl2GEPc2GtdBvw3YZ0NjoLu3Kl1qvA==
ARC-Authentication-Results: i=1;
        ix1-bv-c7-UniversalistesMTAOut-01.renater.fr;
        none
```

X-Spam: Yes

Attention ! Ce n’est pas parce que Google a déterminé que le message était correct et présentait toutes les garanties concernant sa forme et sa provenance qu’il nous fait confiance. Comme il ne fait pas nécessairement confiance au domaine groupware2.renater.fr ce message sera classé comme spam.