RFC 6647 : Email Greylisting: An Applicability Statement for SMTP

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 18 juin 2012

Date de publication du RFC : Juin 2012

https://www.bortzmeyer.org/6647.html

La lutte contre le spam est un domaine où les certitudes (souvent assénées avec conviction) sont plus fortes que les faits. Prenez n'importe quel forum, en ligne ou ailleurs, où on discute des mesures qui pourraient limiter les conséquences du spam, vous y lirez et entendrez des affirmations radicales, par exemple sur l'efficacité du "*greylisting*", sans que les affirmateurs ne se soucient d'appuyer leurs déclarations sur des mesures concrètes, faites dans la réalité. Comme le note notre RFC 6647 [1], le "*greylisting*" est largement déployé, depuis de nombreuses années, marche bien et n'a pas d'inconvénient grave, mais il reste rejeté, sans raisons valables, par de nombreux « experts ». Ce RFC a mis du temps à sortir (il est courant à l'IETF que le texte de la norme vienne très longtemps après le déploiement) mais il changera peut-être cette perception.

Faites l'expérience vous-même : sur un MTA typique qui n'a pas encore le "*greylisting*", activez-le, vous stopperez entre le tiers et les deux tiers du spam avant même que le premier octet de données n'ait été transmis. Si vous souhaitez, avant ou après, vous informer en détail, vous pouvez lire cet excellent RFC 6647, « Tout ce qu'il faut savoir sur le "*greylisting*" ». Il documente enfin officiellement une pratique courante depuis de nombreuses années.

Le principe du "*greylisting*" peut se résumer ainsi : on va refuser temporairement l'accès au serveur de messagerie à certains clients. Les clients normaux réessaieront, ce qu'une partie des spammeurs ne feront pas. On en sera ainsi débarrassé. La différence entre les mises en œuvre du "*greylisting*" se situe essentiellement dans les critères utilisés pour rejeter, puis pour accepter, un client SMTP.

L'idée est donc de considérer qu'un client « nouveau » est soumis au "*greylisting*" jusqu'à ce qu'il devienne « connu ». Une fois connu, on pourra le juger, par exemple sur les messages qu'il envoie. Ces

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6647.txt

mécanismes de réputation ne prenant pas en compte les nouveaux arrivants, le "*greylisting*" sert à les gérer en attendant. Sur la base de ce concept très général, le "*greylisting*" tel que déployé actuellement, consiste à renvoyer un code d'erreur SMTP, indiquant une erreur temporaire (code commençant par 4), à tout nouveau client SMTP. Les spammeurs utilisent souvent des logiciels spéciaux qui, pour de bonnes raisons, ne retentent pas de délivrer le message en cas d'erreur temporaire. Leur renoncement diminue la quantité de spam reçue. Les MTA ordinaires mettent le message dans une file d'attente, puis réessaient (c'est une fonction de base du courrier électronique, on essaie jusqu'à ce que ça passe) et verront donc leur message délivré car, au bout d'un moment, le "*greylisteur*" les considérera comme connus.

Plongeons maintenant dans les détails, après avoir relu les RFC 5322 et RFC 5598.

Il y a plusieurs types de "*greylisting*" et la section 2 en donne une liste. D'abord, ceux qui décident uniquement sur la base des informations de connexion (l'adresse IP et, peut-être, le nom obtenu par une requête de traduction d'adresse en nom). Le "*greylisting*" garde ensuite en mémoire les adresses des sources de courrier. Il refuse (code SMTP 421 ou équivalent) tant qu'une source n'a pas été vue depuis au moins N minutes (avec N variant typiquement entre 10 et 30).

En général, l'information est automatiquement retirée de la base lorsqu'on n'a pas vu cette source depuis X jours (typiquement entre 15 et 60). L'émetteur occasionnel devra donc montrer patte blanche à nouveau.

Naturellement, ce mécanisme peut être combiné avec une liste noire d'adresses IP (n'imposer le "*greylisting*" qu'à celles qui sont sur la liste noire) ou une liste blanche (dispenser du "*greylisting*" celles qui sont sur la liste blanche).

Avec le logiciel Postgrey, on peut avoir ce comportement (ne prendre en compte que l'adresse IP, pas le `MAIL FROM` et le `RCPT TO`), en utilisant l'option `--auto-whitelist-clients` (l'idée est que, si un MTA réessaie plusieurs fois, il ne sert à rien de le tester pour chaque adresse, il a montré qu'il était un ≪ vrai ≫ MTA.)

En laissant le client SMTP aller plus loin, et continuer la session, on peut récolter des informations supplémentaires, qui peuvent servir de paramètres au processus de décision du "*greylisteur*". C'est ainsi qu'on peut attendre le salut SMTP (`EHLO` ou bien le vieux `HELO`) et utiliser alors comme identificateur de la source SMTP le couple {adresse IP, nom annoncé par `EHLO`}. Cela permet, par exemple, de différencier deux clients SMTP situés derrière le même CGN et ayant donc la même adresse IP publique. Autrement, toutes les machines derrière un routeur NAT bénéficieraient du passage de la première machine.

En continuant, on peut aussi récolter l'adresse de courrier émettrice (commande SMTP `MAIL FROM`) et la ou les adresses de courrier de destination (`RCPT TO`). Un "*greylisteur*" qui est allé jusque là peut donc utiliser le triplet {adresse IP, expéditeur, destinataire}. C'est la politique la plus courante. C'est par exemple ce que fait par défaut Postgrey, déjà cité, qui journalise ces informations (voir l'exemple plus loin) pour pouvoir analyser ce qui s'est passé.

Voici d'ailleurs un exemple de session SMTP. Le client est nouveau et donc "*greylisté*" :

```
% telnet mail.bortzmeyer.org smtp
Trying 2001:4b98:dc0:41:216:3eff:fece:1902...
Connected to mail.bortzmeyer.org.
Escape character is '^]'.
220 mail.bortzmeyer.org ESMTP Postfix
```

```
EHLO toto.example.net
250-mail.bortzmeyer.org
250-PIPELINING
250-SIZE 10000000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM:<foo@example.net>
250 2.1.0 Ok
RCPT TO:<stephane+blog@bortzmeyer.org>
450 4.7.1 <stephane+blog@bortzmeyer.org>: Recipient address rejected: Greylisted by greyfix 0.3.9, try again in
```

Cinq minutes après, le client réessaie et est accepté :

```
% telnet mail.bortzmeyer.org smtp
Trying 2001:4b98:dc0:41:216:3eff:fece:1902...
Connected to mail.bortzmeyer.org.
Escape character is '^]'.
220 mail.bortzmeyer.org ESMTP Postfix
EHLO toto.example.net
250-mail.bortzmeyer.org
250-PIPELINING
250-SIZE 10000000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM:<foo@example.net>
250 2.1.0 Ok
RCPT TO:<stephane+blog@bortzmeyer.org>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
```

En pratique, la grande majorité des logiciels de "*greylisting*" ne va pas plus loin. Après tout, le but est de rejeter du spam sans consommer de ressources et donc en réduisant la durée de la session et la capacité réseau utilisée. Toutefois, le RFC cite la possibilité d'attendre encore un peu et d'aller jusqu'à la réception du message (commande SMTP `DATA`). Le "*greylisteur*" ne va pas forcément analyser le corps du message, il peut aussi attendre le `DATA` car certains clients bogués réagissent mal lorsque le refus de continuer la session arrive avant `DATA`. Mais le RFC note que cette analyse du contenu du message reste possible, bien que je ne connaissais pas de "*greylisteur*" qui le fasse. Mais Gabriel Kerneis m'en a cité un : sa-exim <http://marc.merlins.org/linux/exim/sa.html> qui documente cette fonction <http://marc.merlins.org/linux/exim/files/sa-exim-cvs/README.greylisting>. Et il y a aussi smtpf <http://www.snertsoft.com/smtp/smtpf-2.0/smtpf-cf.html#smtpf_grey_list>.

Naturellement, un "*greylisteur*" peut inclure d'autres informations dans sa prise de décision comme la présence de l'expéditeur dans une liste noire (RFC 5782) ou comme un motif dans le nom du client (on fait une requête pour traduire l'adresse IP en nom, et si "*dyn*" ou "*adsl*" sont dans le nom, on rejette temporairement).

Le "*greylisteur*" peut aussi avoir une liste d'exceptions, pour gérer les cas particuliers, par exemple ceux dont on sait qu'ils ne réessaient pas correctement, ou qui attendent trop longtemps avant de réessayer. Par exemple, pour Postgrey, on trouve dans le fichier `/etc/postgrey/whitelist_clients` une liste de domaines dont l'émetteur de courrier est défaillant (il ne réessaie pas) mais dont on souhaite recevoir les messages, par exemple :

```
...
# greylisting.org: Southwest Airlines (unique sender, no retry)
southwest.com
...
```

Comme toute technique de sécurité, le "*greylisting*" est un compromis : il a des bénéfices et il a des coûts. La section 3 est consacrée à l'examen de ces avantages et inconvénients (et la section 9 revient ensuite sur certains de ces choix). L'avantage principal est qu'une partie significative du "*spamware*" ne réessaie jamais, après le refus initial, et que la quantité de spam à traiter va donc diminuer.

L'inconvénient principal est que certains messages seront retardés. Le MTA typique réessaie par défaut au bout de 15 minutes mais c'est réglable et les gros hébergeurs de courrier augmentent en général cette durée, souvent jusqu'à 60 minutes, pour limiter la charge de leurs serveurs, dont la file d'attente est souvent très grosse. Pire, certains (comme le serveur de Southwest Airlines, cité plus haut) ne réessaient pas du tout. Dans ce dernier cas, ils sont en violation caractérisée de la norme SMTP et auront des problèmes avec bien d'autres choses que le "*greylisting*" (une coupure réseau, et hop, le courrier est fichu).

Ce problème du retard est-il sérieux ? Le courrier électronique a toujours été « au mieux », c'est-à-dire qu'une distribution immédiate ne fait pas partie du cahier des charges. Si des gens comptaient sur une délivrance instantanée, c'est qu'ils n'avaient pas compris le principe du courrier.

Le retard est de toute façon rare en pratique. Comme le "*greylisteur*" se souvient des émetteurs corrects (qui réessaient après la première rebuffade) qu'il a rencontrés, et comme la plupart des messages entrants viennent de correspondants réguliers, peu de messages sont vraiment retardés. Si on souhaite à tout prix réduire encore ce nombre, notamment en mettant en service un nouveau serveur, dont la mémoire est vide, on peut toujours envisager de pré-remplir la mémoire avec des adresses extraites du journal (le RFC cite cette possibilité, que je trouve bien compliquée et n'en valant pas la peine).

La section 4 décrit en détail les conséquences non souhaitées du "*greylisting*". Comme pour les notices des médicaments qui préviennent des effets secondaires possibles, il faut la lire en mettant ces inconvénients en rapport des avantages (voir aussi la section 9.1, sur les compromis). Exemple de problème, si on a plusieurs serveurs de courrier (plusieurs entrées dans l'enregistrement MX), et s'ils ne partagent pas leur base d'émetteurs, un client SMTP qui, refusé par le premier, essaiera le second, puis le troisième, etc, sera ainsi retardé plusieurs fois. En parlant de MX secondaire, le RFC recommande également de configurer le primaire pour qu'il accepte les messages des secondaires sans "*greylisting*" (puisque le message a déjà été "*greylisté*" par le secondaire, a priori une machine de confiance).

Autre cas pathologique, celui d'une machine qui n'envoie du courrier que rarement. Ses informations expireront de la base avant le prochain envoi et elle sera donc souvent considérée comme « nouvelle » et donc "*greylistée*".

Outre le retard, un effet parfois ennuyeux du "*greylisting*" est le réordonnancement des messages. Si Alice écrit à Bob et Charlie et que son émetteur est connu du MTA de Bob, mais pas de celui de Charlie,

la délivrance à Charlie sera retardée. Si Bob répond à Alice, et que le MTA de Bob est connu de celui de Charlie, Charlie recevra la réponse de Bob avant le message original d'Alice. Une note personnelle au passage : cela montre l'importance des MUA qui remettent les messages dans l'ordre comme mutt avec son option `set sort=date-sent` (qui dépend d'un réglage correct de la date par l'expéditeur) ou `set sort=threads` (qui utilise les en-têtes du message comme `References:`).

Il y a aussi des problèmes qui sont de la faute du client SMTP. S'il ne réessaie pas, il aura des problèmes avec le "*greylisting*" mais aussi avec toute panne réseau temporaire. S'il considère tous les codes d'erreur SMTP comme permanentes (alors que la section 4.2.1 du RFC 5321 est bien clair que les codes de type 4xy sont des erreurs temporaires), il aura des problèmes avec le "*greylisting*" et avec certaines erreurs sur le MTA.

Si le client SMTP utilise une adresse d'expéditeur de courrier différente pour chaque essai (une idée très baroque mais tout existe sur l'Internet), et que le "*greylisteur*" utilise cette adresse pour déterminer si un envoi est nouveau, alors, le message ne passera jamais.

Enfin, les logiciels de test des adresses de courrier peuvent avoir des problèmes avec le "*greylisting*" comme Zonecheck `<http://www.zonecheck.fr/>` et sa bogue #9544 `<https://savannah.nongnu.org/bugs/index.php?9544>`.

Après cette liste de problèmes possibles, les recommandations concrètes, pour que le "*greylisting*" apporte joie et bonheur et que les problèmes restent virtuels. Elles figurent en section 5 et les principales, tirées de la longue expérience du "*greylisting*", sont notamment :

— Mettre en œuvre le "*greylisting*" en utilisant comme tuple définissant un visiteur le triplet {adresse IP source, adresse de courrier en `MAIL FROM`, première adresse de courrier dans le `RCPT TO`}. L'inclusion des adresses de courrier a notamment pour but de limiter le réordonnancement des messages. Une fois que ce test a été passé une fois, autoriser l'adresse IP quelles que soient les adresses (le MTA client a montré qu'il réessayait, de toute façon).
— Permettre à l'administrateur système de configurer la période de quarantaine imposée au nouveau client. Le RFC définit une valeur par défaut souhaitable comme étant entre 1 minute et 24 heures (oui, c'est large). (Avec Postgrey, c'est l'option `--delay`.)
— Supprimer les entrées de la base au bout d'une période d'inactivité configurable (par défaut, d'au moins une semaine, dit le RFC). (Avec Postgrey, c'est l'option `--max-age`.)
— Considérer comme adresse IP de la source un préfixe entier (par exemple un /24 pour IPv4), pas juste une adresse, pour le cas des "*pools*" de serveurs SMTP (le second essai ne viendra pas forcément de la même machine). Cela réduit également la taille de la base nécessaire, ce qui peut être très important pour IPv6 (cf. section 7, ainsi que la section 9.2) où le nombre d'adresses potentielles est bien plus grand. (Postgrey a une option limitée, `--lookup-by-subnet`, mais où la longueur du préfixe est fixe, 24 bits en IPv4. Greyfix est plus souple.)
— Permettre de définir une liste blanche d'adresses autorisées à passer outre. (Chez Postgrey, ce sont les fichiers de configuration comme `/etc/postgrey/whitelist_clients`.)
— Ne pas faire de "*greylisting*" sur le service de soumission de courrier (RFC 6409).

En revanche, notre section 5 ne recommande pas de valeur particulière pour le code de retour SMTP lorsqu'on refuse temporairement un client. 421 et 450 semblent les deux choix possibles (RFC 5321, section 4.2.2).

SMTP permet de mettre dans la réponse, outre un code numérique, un court texte. Faut-il dire explicitement que du "*greylisting*" est en action, ce qui peut aider un spammeur ? Le RFC estime que oui, pour être sympa avec les utilisateurs légitimes (c'est ce que font les deux logiciels de "*greylisting*" montrés en exemple dans cet article).

La section 6 du RFC est consacrée à la question de la mesure de l'efficacité du "*greylisting*". Elle suggère une expérience intéressante que, à ma connaissance, personne n'a tenté : regarder si les adresses

IP "*greylistées*" (et qui ne réessaient pas) sont plus souvent présentes que les autres dans les grandes DNSBL. Une corrélation indiquerait qu'on gêne bien surtout les spammeurs.

Il existe aujourd'hui des mises en œuvre du "*greylisting*" pour à peu près tous les MTA possibles. Pour Postfix, les deux solutions les plus courantes en logiciel libre sont Greyfix <http://www.kim-minh.com/pub/greyfix/> et Postgrey <http://postgrey.schweikert.ch/>. On configure Greyfix en indiquant dans le `master.cf` de Postfix :

```
greyfix    unix  -       n       n       -       -       spawn
  user=nobody argv=/usr/local/sbin/greyfix --greylist-delay 300 --network-prefix 28
```

Les deux options indiquées ici lui imposent de ne pas accepter un nouveau client avant 300 secondes, et de considérer toutes les adresses d'un même /28 comme équivalentes. (Greyfix peut aussi le faire en IPv6, avec `--network6-prefix`.). Un nouveau client rejeté est journalisé ainsi :

```
May 29 14:59:32 aetius postfix/smtpd[25150]: NOQUEUE: reject: RCPT from unknown[117.214.206.76]:4974: 450 4
```

Pour Postgrey, qui tourne comme un démon séparé, ici sur le port 10023, la configuration se fera dans le `main.cf` :

```
smtpd_recipient_restrictions = permit_mynetworks, \
           check_policy_service inet:127.0.0.1:10023
```

Et un client rejeté produira :

```
May 28 16:08:33 lilith postfix/smtpd[20582]: NOQUEUE: reject: RCPT from 241.146.12.109.rev.sfr.net[109.12.14
May 28 16:08:33 lilith postgrey[1447]: action=greylist, reason=new, client_name=241.146.12.109.rev.sfr.net,
```

Une question un peu philosophique, maintenant, que ne traite guère notre RFC. Pourquoi est-ce que les logiciels de spam courants ne réessaient pas, après l'erreur temporaire reçue ? Cela peut sembler déroutant puisque, de leur point de vue, ce serait la solution évidente au "*greylisting*". Il y a plusieurs raisons à cette passivité. L'une est que le "*greylisting*" n'est pas universellement déployé. S'il l'était, les choses changeraient peut-être (finalement, était-ce une bonne idée de publier ce RFC ?) Ce point de vue est par exemple développé dans l'"*Internet-Draft*" `draft-santos-smtpgrey`. Ensuite, les spammeurs visent typiquement le volume, pas la qualité. Pourquoi s'embêter à réessayer alors qu'il est bien plus efficace, en terme de volume, de passer tout de suite à la victime suivante dans la liste ? Mais il y a aussi sans doute aussi crainte d'être repérés s'ils essaient : un zombie essaie en général d'être discret.

De bonnes lectures :

- L'une des toutes premières publications sur le "*greylisting*" dans SAUCE <http://www.gnu.org/software/sauce>, en 2001.
- Le premier article détaillé, « "*The Next Step in the Spam Control War : Greylisting*" <http://projects.puremagic.com/greylisting/whitepaper.html> » en 2003.
- Bien plus récent, mon article <https://www.bortzmeyer.org/greylisting.html> qui date de trois ans.
- Et le site « officiel <http://www.greylisting.org/> ».

Merci à Kim-Minh Kaplan pour sa relecture.