Rspamd

Rspamd est plus qu'un simple antispam : il s'occupera aussi d'ajouter les signatures DKIM et ARC à vos mails sortants et pourra faire la liaison avec un antivirus. C'est un tout-en-un vraiment sympa □□

Installation

Je vous laisse aller voir ça sur le site de rspamd.

Création et utilisation de clés DKIM et ARC

Création

NB : par défaut, Rspamd va chercher les clés dans le dossier `/var/lib/rspamd/dkim/`. Cependant, je préfère les mettre dans le dossier `/etc/dkim` : on pense plus souvent à sauvegarder `/etc` que `/var/lib/rspamd/`.

```
rspamadm dkim_keygen -k /etc/dkim/example.com.dkim.key -b 2048 -s 'dkim' -d example.com > /etc/dkim/example.com.dkim.txt
```

- `-k` => fichier qui contiendra la clé
- `-b` => nombre de bits de la clé (défaut : 1024)
- `-s` => nom du sélecteur (voir plus bas)
- `-d` => le domaine à signer

À noter, la redirection de la sortie de la commande vers `/etc/dkim/example.com.dkim.txt` : ce fichier contient l'enregistrement DNS que vous devrez créer pour votre domaine pour déclarer la clé DKIM utilisée.

Cela ressemble à :

```
dkim._domainkey IN TXT ( "v=DKIM1; k=rsa; "

"p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzgAF2ozDnleUGRbtwmbTEg1zmmoSLhOjsT96q0P+JOrTPnG
X/oIWwx2MkTRW46gSU7Ya1ByG9EKfEQo3V+Zfr5xeY+O0ksl8nrHUk56haW7kqVAEhyo4NPqhhTRUheAIMgLbyYlFNOqpQ
DCdmfyn6fvObK6caqtNXAWy3vWTeMacBgx1JGfrYE1NFyNqKcfHcbtXXfSGNo6phVz9K"

"1Tll3wvZhdW3hBwgq49cZ5yp0IsrlLOfqMOnHcS83YHlNMRVVGvPko8+ucMhKktbAoDdEMMWupxyWGs1M1xKW0RQxFyYi
5oZhSTW53VpyzldrlWXInerDRW2hn1amA2dlWwewIDAQAB"
) ;
```

Le sélecteur est une clé qui servira, on le voit, dans le nom de l'enregistrement DNS. Il sera indiqué dans l'en-tête de signature DKIM des mails, et donc utilisé par les antispams pour aller chercher le bon enregistrement DNS qui déclare la clé utilisée. Par défaut, pour DKIM, rspamd utilise le sélecteur `dkim` et je ne vois pas de raison d'en changer (en plus ça ferait des modifications de configuration supplémentaires).

Le sélecteur est aussi utilisé par défaut par rspamd pour choisir la clé à utiliser pour signer les mails.

Pour les clés ARC, c'est tout pareil, mais on change le sélecteur pour `arc`. Vous pouvez utiliser le même dossier `/etc/dkim`, c'est ce que je fais.

Utilisation par Rspamd

Si vous avez utilisé `/var/lib/rspamd/dkim/` (et `/var/lib/rspamd/arc` pour ARC) comme chemins à la place de `/etc/dkim`, vous n'avez rien à faire : rspamd cherche les clés des domaines avec le chemin `/var/lib/rspamd/dkim/$domain.$selector.key` (et `/var/lib/rspamd/arc/$domain.$selector.key` pour ARC).

Si comme moi vous utilisez `/etc/dkim` pour ranger vos clés, il va falloir surcharger la configuration de rspamd.

Créez les fichiers `/etc/rspamd/local.d/dkim_signing.conf` et `/etc/rspamd/local.d/arc.conf` et mettez-y ceci :

```
path = "/etc/dkim/$domain.$selector.key"
```

Relancez rspamd, et c'est normalement tout bon □□

Ajouter les en-têtes donnant le spam score dans les mails

Certains spams passent, des mails légitimes ne passent pas… pour comprendre ça, on peut aller dans les logs pour y retrouver les infos qui vont bien, ou alors on ajoute directement ces informations dans les en-têtes des mails □□

Dans le fichier `/etc/rspamd/local.d/milter_headers.conf`, mettez :

```
extended_spam_headers = true;
```

Relancez rspamd, et c'est normalement tout bon □□

Se prémunir du phishing

Rspamd est capable d'utiliser les listes d'URL de phishing d'OpenPhish et PhishTank.

L'utilisation de PhishTank est activée de base dans `/etc/rspamd/modules.d/phishing.conf` (sur la version des dépôts officiels de rspamd, tout du moins).

Pour utiliser OpenPhish :

```
echo 'openphish_enabled = true;' >> /etc/rspamd/local.d/phishing.conf
systemctl reload rspamd.service
```

Notez que Rspamd est capable de gérer la liste premium d'OpenPhish (qui contient plus d'URL).

On peut aussi utiliser un fichier local contenant une liste d'adresses malveillantes.

Pour cela :

```
cat <<EOF >> /etc/rspamd/local.d/phishing.conf
generic_service_enabled = true;
generic_service_name = 'PhishStats';
generic_service_symbol = "PHISHED_PHISHSTATS";
generic_service_map = "file:///opt/phishstats_urls.txt";
EOF
```

Je prends ici les adresses de PhishStats. Pour construire le fichier (le site fourni un CSV, inutilisable, donc) :

```
curl -s https://phishstats.info/phish_score.csv |
    grep -v "^#" |
    cut -f 3 -d ',' |
    tr -d '"' > /tmp/phishstats_urls.txt
if [[ $(wc -l /tmp/phishstats_urls.txt | cut -f 1 -d ' ') -gt 0 ]]; then
    mv /tmp/phishstats_urls.txt /opt
    systemctl reload rspamd.service
fi
```

Il suffit de mettre ce bout de code dans un script et d'appeler ce script régulièrement pour mettre à jour les adresses de PhishStats (le site dit que le fichier est mis à jour toutes les 90 minutes).

Il faut encore définir un poids pour le symbole qu'on vient d'ajouter. Éditez `/etc/rspamd/local.d/phishing_group.conf`

```
symbols {
    "PHISHED_PHISHSTATS" {
        weight = 7.0;
        description = "Phished URL";
        one_shot = true;
    }
}
```

Et relancez rspamd :

```
systemctl reload rspamd.service
```

Comprendre les symboles Rspamd

Dans les en-têtes ajoutés dans les mails via la configuration juste au-dessus, il y a les symboles rspamd. Ce sont différentes catégories de vérification antispam, avec un score. C'est la somme de ces scores qui donne le spam score qui va déclencher l'acceptation du mail, son classement en spam ou carrément son refus.

Cependant ces symboles n'ont pas forcément une signification évidente. Voici une liste de symboles expliqués (cette liste n'est pas exhaustive) :

- ARC_REJECT : la signature ARC est-elle valide ?
- ARC_SIGNED : existe-t-il une signature ARC ?
- ASN : score de l'IP par rapport à son ASN auquel il appartient. Rspamd fait des statistiques au niveau des adresses IP, sous-réseaux, ASN et pays
- BAYES_SPAM : analyse bayésienne du mail
- CTYPE_MIXED_BOGUS : mails `multipart/mixed` sans partie non-textuelle
- DKIM_SIGNED : le message possède une signature DKIM (sans préjuger de sa validité)
- DKIM_TRACE : un truc avec DKIM, c'est sûr, mais je sais pas quoi exactement
- DMARC_POLICY_SOFTFAIL : la vérification DMARC a échoué
- FORGED_RECIPIENTS : les destinataires ne sont pas les mêmes que la commande mail `RCPT TO`
- FORGED_RECIPIENTS_MAILLIST : les destinataires ne sont pas les mêmes que la commande mail `RCPT TO` mais le message vient d'une liste de diffusion
- FORGED_SENDER : l'en-tête `Sender` est forgé (différence entre l'en-tête `From` et `MAIL FROM`)
- FORGED_SENDER_MAILLIST : l'en-tête `Sender` est forgé (différence entre l'en-tête `From` et `MAIL FROM`) mais le message vient d'une liste de diffusion
- FROM_NEQ_ENVFROM : l'adresse `From` est différente de celle de l'enveloppe
- FROM_NO_DN : l'en-tête `From` n'a pas de *display name*
- HAS_LIST_UNSUB : possède l'en-tête `List-Unsubscribe`
- HAS_REPLYTO : est-ce que le mail a bien un header `Reply-To` ?
- LOCAL_WL_IP : vérification de la liste blanche locale
- MAILLIST : le mail semble venir d'une liste de diffusion
- MID_RHS_MATCH_FROM : est-ce qu'on retrouve l'adresse `From` dans le `Message-ID` ?
- MID_RHS_NOT_FQDN : le `Message-ID` ne contient pas de nom de domaine pleinement qualifié (*fqdn*)
- MIME_GOOD : `Content-Type` connu
- MIME_HTML_ONLY : pas de version texte du message HTML
- MIME_TRACE : un truc qui a à voir avec les types MIME, mais je sais pas quoi exactement
- MV_CASE : l'en-tête `MIME-Version` n'a pas la bonne casse (ex : `Mime-Version`)
- ONCE_RECEIVED : il n'y a qu'un seul en-tête `Received`, ce qui peut indiquer une machine compromise (d'après la doc de rspamd)
- PRECEDENCE_BULK : envoi de mail en masse
- RCPT_COUNT_ONE : un seul destinataire
- RCVD_COUNT_THREE : le mail a entre 3 et 5 en-tête `Received` (a transité par 3/4/5 serveurs différents)
- RCVD_IN_DNSWL_FAIL : fail du test [[https://www.dnswl.org]] (une liste blanche d'adresses IP)
- RCVD_TLS_LAST : le dernier serveur (*last hop*) utilise un transport sécurisé
- R_DKIM_ALLOW : DKIM correct
- RECEIVED_SPAMHAUS_FAIL : a priori, blacklisté chez Spamhaus (une RBL)
- R_EMPTY_IMAGE : le message contient des parties texte vides et une image

- REPLYTO_DN_EQ_FROM_DN : le *display name* de l'en-tête `Reply-To` est-il le même que celui du `From` ?
- REPLYTO_DOM_NEQ_FROM_DOM : le domaine `Reply-To` ne correspond pas à celui de `From`
- R_SPF_ALLOW : respect de l'enregistrement SPF
- TO_DN_NONE : Aucun des destinataires n'a de *display names*
- TO_DOM_EQ_FROM_DOM : le domaine `To` est le même que celui de `From`

Mettre des domaines en liste d'autorisation pour les vérifications des SURBL

Contexte : vous avez un domaine qui se retrouve dans une SURBL (une liste noire de domaines de phishing/spam/etc). Problème : si quelqu'un souhaite vous envoyer un mail d'abuse à propos de ce domaine sans le protéger (genre en n'écrivant pas `[https]://lstu [.] fr`,) ça arrive dans vos spams et vous ne voyez pas les abuses.

Pour éviter ce problème, vous pouvez mettre des exceptions pour les vérifications des SURBL.

Mettez simplement vos domaines dans `/etc/rspamd/local.d/maps.d/surbl-authorized_list.inc.local` (un domaine par ligne) et rechargez `rspamd`.

Exemple de fichier `/etc/rspamd/local.d/maps.d/surbl-authorized_list.inc.local` :

```
lstu.fr
```

Augmenter le score de spam des mails à destination d'une certaine adresse mail

Exemple d'usage : comme je publie des modules Perl sur le CPAN, j'ai une adresse @cpan.org qui a été automatiquement créée, dont les mails sont transférés chez moi, et qui se retrouve spammée à longueur de journée. Comme il y a quand même une possibilité d'avoir des mails légitimes dessus, je ne la bloque pas complètement. Par contre, augmenter le score de spam permet de faire passer

des mails qui ont déjà un petit score de spam dans la catégorie « Oui, c'est bien du spam ».

C'est le module `multimap` qui s'occupe de ça (voir la documentation).

Créer le fichier `/etc/rspamd/local.d/multimap.conf` :

```
cpanmail_to {
    type = "header";
    header = "Delivered-To";
    filter = "email:addr";
    map = "file:///etc/rspamd/local.d/cpan_map";
    symbol = "CPAN_DELIVERED_TO";
    description = "Delivered-To is ldidry@cpan.org";
    score = 5.0;
}
```

Le score à ajouter dépend bien évidemment des seuils que vous avez réglés dans rspamd.

Créer le fichier `/etc/rspamd/local.d/cpan_map` :

```
ldidry@cpan.org
```

Redémarrer `rspamd` et profiter d'une boîte mail avec moins de spam

Forcer une politique DMARC

Quand la politique DMARC n'est pas respectée, ça influence le score de spam, mais ça ne rejette pas forcément le mail. Pour forcer la politique appliquée selon ce que recommande l'enregistrement DMARC du domaine du mail, mettre ceci dans `/etc/rspamd/local.d/dmarc.conf` (adaptez selon vos envies, bien évidemment) et redémarrer `rspamd` :

```
actions = {
    quarantine = "add_header";
    reject = "reject";
}
```

Modifier le score d’un mail selon le langage détecté

Tiré de https://github.com/postalserver/postal/discussions/1754

Mettre ceci dans `/etc/rspamd/local.d/lang_filter.lua` :

```
local rspamd_logger = require 'rspamd_logger'

local deny_langs = {
    ['zh'] = true,
    ['ru'] = true,
}

rspamd_config:register_symbol{
    type = 'normal',
    name = 'LANG_FILTER',
    score = 6.0,
    group = 'LANG_FILTER',
    description = 'Deny languages',
    flags = 'fine',
    callback = function(task)
        local any_ok = false
        local parts = task:get_text_parts() or {}
        local ln
        for i,p in ipairs(parts) do
            ln = p:get_language() or ''
            local dash = ln:find('-')
            if dash then
                -- from zh-cn to zh
                ln = ln:sub(1, dash-1)
            end

            if deny_langs[ln] then
                rspamd_logger.infox("lang for %1 is %2 -DENY", i, ln)
            else
                any_ok = true
```

```
                rspamd_logger.infox("lang for %1 is %2 -OK", i, ln)
                break
            end
        end
        if any_ok or not ln or #ln == 0 then
            return false
        else
            return true
        end
    end,
}
```

Mettre ceci dans `/etc/rspamd/local.d/rspamd.lua` :

```
local local_conf = rspamd_paths['LOCAL_CONFDIR']

-- filtrer selon le langage
dofile(local_conf .. '/local.d/lang_filter.lua')
```

Pour faire l'inverse et n'autoriser que certains langages, voir https://rspamd.com/doc/lua/examples.html#languages-filter.

Envoyer des rapport DMARC

Cela se fait avec la commande `rspamadm dmarc_report`, à mettre dans une tâche cron, mais il faut un peu de configuration supplémentaire.

Mettre ceci dans `/etc/rspamd/local.d/dmarc.conf` :

```
reporting {
    # Required attributes
    enabled = true; # Enable reports in general
    email = 'dmarc_reports@example.com'; # Source of DMARC reports
    domain = 'example.com'; # Domain to serve
    org_name = 'Example organisation'; # Organisation
    # Optional parameters
    bcc_addrs = ["postmaster@example.com"]; # additional addresses to copy on reports
    report_local_controller = false; # Store reports for local/controller scans (for testing
only)
```

```
    helo = 'rspamd.localhost'; # Helo used in SMTP dialog
    smtp = '127.0.0.1'; # SMTP server IP
    smtp_port = 25; # SMTP server port
    from_name = 'Rspamd'; # SMTP FROM
    msgid_from = 'rspamd'; # Msgid format
    max_entries = 1k; # Maxiumum amount of entries per domain
    keys_expire = 2d; # Expire date for Redis keys
    #only_domains = '/path/to/map'; # Only store reports from domains or eSLDs listed in this
map
    # Available from 3.3
    #exclude_domains = '/path/to/map'; # Exclude reports from domains or eSLDs listed in this
map
    #exclude_domains = ["example.com", "another.com"]; # Alternative, use array to exclude
reports from domains or eSLDs
}
```

Il est aussi nécessaire d'avoir un serveur redis et de configurer le module DMARC pour l'utiliser.

Mettre ceci dans `/etc/rspamd/local.d/dmarc.conf` :

```
servers = "127.0.0.1";
```

Révision #24
Créé 2020-01-23 16:15:16 CET par Luc
Mis à jour 2024-06-30 17:14:21 CEST par Luc