Cours 3: Mail

DNS

Enregistrement MX, typiquement

```
@                   MX      10 monserveur1
@                   MX      10 monserveur2
@                   MX      20 monserveur-fallback
monserveur1         A       192.168.56.10
monserveur2         A       192.168.56.11
monserveur-fallback A       192.168.42.1
```

Indication de priorité pour ne pas solliciter le fallback en temps normal.

SMTP, SUBMISSION

SMTP (Simple Mail Transport Protocol)

Port 25, envoi normal de mail.

Pour un mail à destination de `machin@truc.bidule` * On demande à DNS l'enregistrement MX pour `truc.bidule.` * On se connecte sur le port 25 de l'IP retournée * On pousse le mail

Note: pour éviter autant que possible de laisser les mails passer en clair, presque systématiquement StartTLS est utilisé:

- Le serveur indique qu'il sait chiffrer avec TLS
- Le client demande à basculer en TLS
- Les deux échangent des clés
- La suite du protocole se fait en chiffré.

Pour cela, le serveur a besoin d'avoir sa propre paire de clés publique/privée, en général c'est généré automatiquement à l'installation.

SMARTHOST

En théorie on dirait à son PC d'envoyer ses mails au reste du monde en SMTP.

Mais on préfère utiliser le smarthost de notre FAI:

- Pour qu'il s'occupe de réessayer d'envoyer les mails si le serveur SMTP cible est temporairement en panne.
- Parce que notre FAI ne nous laisse pas nous connecter aux ports 25 du reste du monde (empêcher d'émettre du spam)

Et donc configurer son PC pour utiliser le SMTP de son FAI (toujours sur le port 25).

E.g. `smtp.emi.u-bordeaux.fr`

En théorie, le nom du serveur SMTP utilisable est fourni via DHCP. En pratique c'est rarement configuré pour.

SUBMISSION

C'est donc plus simple d'utiliser le serveur submission de notre fournisseur d'adresse mail (port 587).

On configure son client mail: pour chaque adresse mail qu'on utilise en "From", on configure le serveur submission à utiliser.

On peut se balader dans le monde entier, on envoie ses mails via son fournisseur d'adresse mail (587 exige typiquement une authentification, en chiffré). Il n'accepte que les mails avec le `From` correspondant.

POP3/IMAP

Les mails atterissent sur le serveur de mail. Pour les récupérer, pop3/imap.

- POP3: historique, limité (pas de dossiers)
- IMAP: plus avancé: dossiers, gestion des mails, mode push plutôt que pull (aussi appelé idle)

Chiffrement

Anciennement, pop3 (110) et imap(143), les mails passent en clair, mauvaise idée.

pop3s (995) et imaps (993), connexion entièrement chiffrée. Le serveur a besoin d'avoir sa propre paire de clés publique/privée.

Webmail

C'est une simple sur-couche web ! Derrière, se connecte en imap au serveur de mail

SPF, DMARC

Toujours pour empêcher les spams... On veut vérifier le `From` des mails.

Problème: rien dans SMTP n'empêche d'envoyer un mail avec n'importe quel `From`.

SPF

On ajoute un enregistrement DNS, par exemple:

```
labri.fr.       TXT "v=spf1 ip4:147.210.0.0/16 ip6:2001:660:6101::/48 ~all"
```

Indique qu'en principe les mails `From: *@labri.fr` viennent des IPs du réseau de l'université de Bordeaux. Quand on utilise le port submission depuis l'autre bout d'internet, ça reste bien le cas: le serveur submission qu'on utilise est bien dedans.

`~all` indique que ce n'est pas vraiment normal d'émettre un mail `From: *@labri.fr` depuis ailleurs sur Internet, et donc les anti-spam vont alors marquer en tant que spam.

On pourrait carrément indiquer `-all` pour faire rejeter le mail. Mais le rejet va être retourné au `From`, qui est probablement forgé... Préférer donc `~all` et laisser l'antispam catégoriser.

Mais cela oblige tout le monde à utiliser le serveur submission. Si on ne veut pas forcer cela, on peut mettre `?all` pour indiquer que ce n'est pas anormal. Il se peut que des antispam augmentent quand même un peu le score de spam dans ce cas.

Note: SPF n'est pas une garantie, c'est juste une aide pour catégoriser

DKIM

Pour avoir des garanties, il faut signer électroniquement.

On peut demander au serveur de mail d'ajouter une signature électronique dans le mail.

-> garantie que le mail reçu est bien passé par ce serveur de mail.

Mais encore faut-il savoir quelle clé vérifier -> enregistrement DNS.

Par exemple, si l'on reçoit un mail avec:

```
DKIM-Filter: OpenDKIM Filter v2.11.0 smtp.gnome.org 119442805C16
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=gnome.org;
        s=default; t=1726590069;
        bh=DYBZPHI9nkalm1UzsyY3ZMD6qJjtVXs1480QUQHknok=;
        h=Date:From:Reply-To:To:In-Reply-To:References:Subject:List-Id:
        List-Unsubscribe:From;
        b=x5LmzK8DrWh96uWiJDDNbSfbXGM6sWICEs0MKSZxq1sxh1qsi/3EQiZdsJKP5lhQ8
        xDVz5jAHoCPd7i84hHHLeJAWbrCYs5J1W1NmXS1MVz2toZKyJdUAl151mE9UVaKH3r
        X5vJmcoHxxPgoa1A3OT7P9w1bjim/hKEJmNu6UIc=
```

`s=default` indique le sélecteur. On récupère la clé depuis DNS en ajoutant `_domainkey` et le nom de domaine du `From` du mail:

```
$ dig +short txt default._domainkey.gnome.org
"v=DKIM1; p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDTA1zgW8+e7haZgEoZAMmoMC7jwekFKv6AV7OQbsOKLpAnLivyVUb
```

`p=...` donne la clé publique, on peut l'utiliser pour vérifier la signature du mail.

Mais si on reçoit un mail non signé avec DKIM ?

DMARC

On interroge DNS:

```
$ dig +short txt _dmarc.gnome.org
"v=DMARC1; p=reject; adkim=s; aspf=s;"
```

`gnome.org` a carrément indiqué que la politique (`p`) est de rejeter le mail, en vérifiant strictement (`s`) à la fois DKIM et SPF.

Pour l'utilisateur, cela oblige vraiment à utiliser le serveur submission pour émettre des mails `From: *@gnome.org`.

En résumé, mise en œuvre

- SPF: facile, il suffit d'ajouter le champ DNS pour déclarer ses serveurs de mail
- DKIM: fabriquer une clé pour signer, l'enregistrer dans DNS
- DMARC: une fois qu'on est sûr d'avoir correctement mis en œuvre SPF et DKIM, enregistrer dans DNS qu'ils sont obligatoires.