

# Reverse Proxy

## Présentation

Cet article ne fera référence qu'aux proxy et reverse proxy (proxy inverse) dans le cadre de l'utilisation des protocoles web http et https.

## Définitions

Commençons par le proxy. Les clients http (communément des navigateurs) d'un site (un laboratoire par exemple) passent par une passerelle pour accéder à des serveurs situés sur le web. Les données sont le plus souvent mises en cache permettant ainsi une diminution de la bande passante et des réponses rapides aux requêtes les plus fréquentes des clients. Ces derniers sont alors vus comme une seule adresse IP par les serveurs distants. L'utilisation classique d'un proxy est connue et fréquemment employée. Elle permet aussi de diminuer la nécessité de mettre un poste de travail routé sur le Net et de mettre en place un contrôle éventuel des sites distants accédés. Les clients ont souvent connaissance de la présence du proxy.

Inverser le fonctionnement d'un proxy pour avoir un reverse proxy revient à avoir des clients http sur le web (souvent inconnus) dont les requêtes passent par le proxy pour accéder aux serveurs du site qui héberge aussi celui-ci. Les clients ne voient qu'une adresse IP et n'ont pas connaissance du système mis en place. Le reverse proxy devient alors le point d'entrée unique aux serveurs web d'un site.

## Usages et avantages

L'utilisation d'un reverse proxy permet d'envisager une répartition des applications web par serveur. L'approche idéale est, dans bien des cas, qu'une seule application web soit hébergée sur un serveur web. Mais on peut répartir les applications sur les serveurs en fonction des langages de programmation avec lesquels elles ont été développées. Pour la sécurité et la stabilité du système, il n'est pas souhaitable de mélanger applications et/ou version de langage sur un même et seul serveur.

Le point d'entrée unique qu'est le reverse proxy permet un contrôle des accès aux applications web. Le contrôle de charge est ainsi plus aisé avec une telle architecture réseau. Une répartition de la charge est alors envisageable en fonction des besoins. Cela permet aussi de « reculer » les ressources sensibles d'un site, comme par exemple les bases de données.

Ce point d'entrée unique permet aussi de simplifier l'établissement de règle sur le firewall du site. Cela est pratique lorsque l'on se trouve dépendant d'un hébergeur.

Une déclaration DNS peut suffire mais il est possible d'en avoir plusieurs si le serveur prend en charge les virtual hosts.

Si un cache est utilisé, les performances des pages et des données fréquemment demandées pourront être grandement améliorées.

L'utilisation de ce type de serveur en frontal de serveurs d'applications comme Zope est grandement encouragée par les développeurs de ce dernier. Les performances des serveurs http embarqués, par Zope et d'autres, comme les très à la mode framework MVC, sont très faibles.

## Inconvénients

Le premier inconvénient qui vient immédiatement à l'esprit est que le reverse proxy devient un point central. S'il est en panne tous les serveurs web sont inaccessibles. Une compromission de celui-ci peut avoir un impact fort surtout s'il s'y trouve des données importantes dans le cache. Le serveur doit être idéalement dans une DMZ afin de limiter la portée d'une intrusion, permettant ainsi la mise à l'écart par exemple des bases de données.

La multiplication des machines peut aussi apparaître rédhibitoire. La mise à jour des redirections et des règles de réécritures des URL peuvent apparaître fastidieuses.

L'utilisation d'un reverse proxy rend la politique de contrôle d'accès par IP impossible. Les serveurs situés derrière lui ne voient qu'une adresse IP. Lors d'une migration, il faut donc veiller à remplacer ce genre de contrôle.

Lorsque l'on utilise le protocole https, une rupture de ssl a lieu sur le reverse proxy. Mais le reverse proxy peut permettre d'ajouter https à des serveurs qui en sont dépourvus comme par exemple Zope.

## Mise en œuvre

La démocratisation de la virtualisation permet d'envisager la mise en place d'un tel système sans exploser le budget équipement. Le reverse proxy sera évidemment le plus gourmand en ressource et nécessitera éventuellement un machine physique. Coté réseau, la mise en place d'une DMZ, dans laquelle sera installé le reverse proxy, sera un plus.

## Les logiciels

Un large choix de logiciels est disponible pour mettre en place un serveur reverse proxy. Les serveurs mandataires (proxy classique) ont souvent la possibilité d'en faire. Mais des logiciels ont été développés dans le but spécifique de faire du reverse proxy. Les serveurs web ont aussi souvent la possibilité, par l'intermédiaire de modules optionnels, de rendre ce service.

Pour le choix d'une solution, il faut bien entendu regarder les fonctionnalités du serveur. Parmi celles-ci, virtual host, ssl, cache et réécriture d'URL sont importantes.

Voici la description de quelques solutions possibles parmi les logiciels libres.

À noter que parmi les logiciels propriétaires, Microsoft propose, sur sa plateforme Windows®, Internet Security and Acceleration Server (ISA Server).

## Squid

Squid est un serveur proxy disponible depuis de nombreuses années. Il permet de mettre en place un reverse proxy avec la possibilité de gérer les virtual host depuis la série 3. Il est très stable et rapide avec une bonne gestion du cache. Il a été longtemps préféré à Apache du temps où ce dernier n'était disponible qu'en version 1.3.

## Pound

Le développement de Pound a été initié pour servir de solution frontale aux serveurs d'applications Zope. Il apporte ainsi le support ssl à Zope qui en est dépourvu jusqu'à la version 3. L'équilibre de charge est sa spécialité avec en plus une détection de panne éventuelle d'un des serveurs.

## Apache

Le serveur le plus utilisé sur le Net est bien entendu capable de faire du reverse proxy. Ses

performances sont depuis la version 2 à la hauteur de sa réputation. De plus, sa popularité lui permet d'avoir de très nombreux modules optionnels. Virtual host, ssl, cache et réécritures d'URL tout y est et bien plus.

Des modules très intéressants dans le cadre d'un reverse proxy existent, comme, par exemple, `mod_evasive` pour prévenir d'une attaque de type DOS ou DDOS et, surtout, **`mod_security`** qui surveille et protège les applications web situées derrière ce module. Il fonctionne comme un filtre basé sur un système de règles relativement simples à écrire. Le filtre s'applique sur toute requête en fonction de patterns prédéfinis ou définis par l'utilisateur. À noter que ce module peut très bien être utilisé sans la mise en place d'un reverse proxy.

## Varnish

Le petit dernier qui monte, se décrit comme un accélérateur http. Il entre donc, plutôt, dans la catégorie reverse proxy développé pour cela. Il est maintenant conseillé (avec Apache) par la documentation de Zope (avant c'étaient Apache et Squid) et il se gratifie d'excellentes performances. À tester.