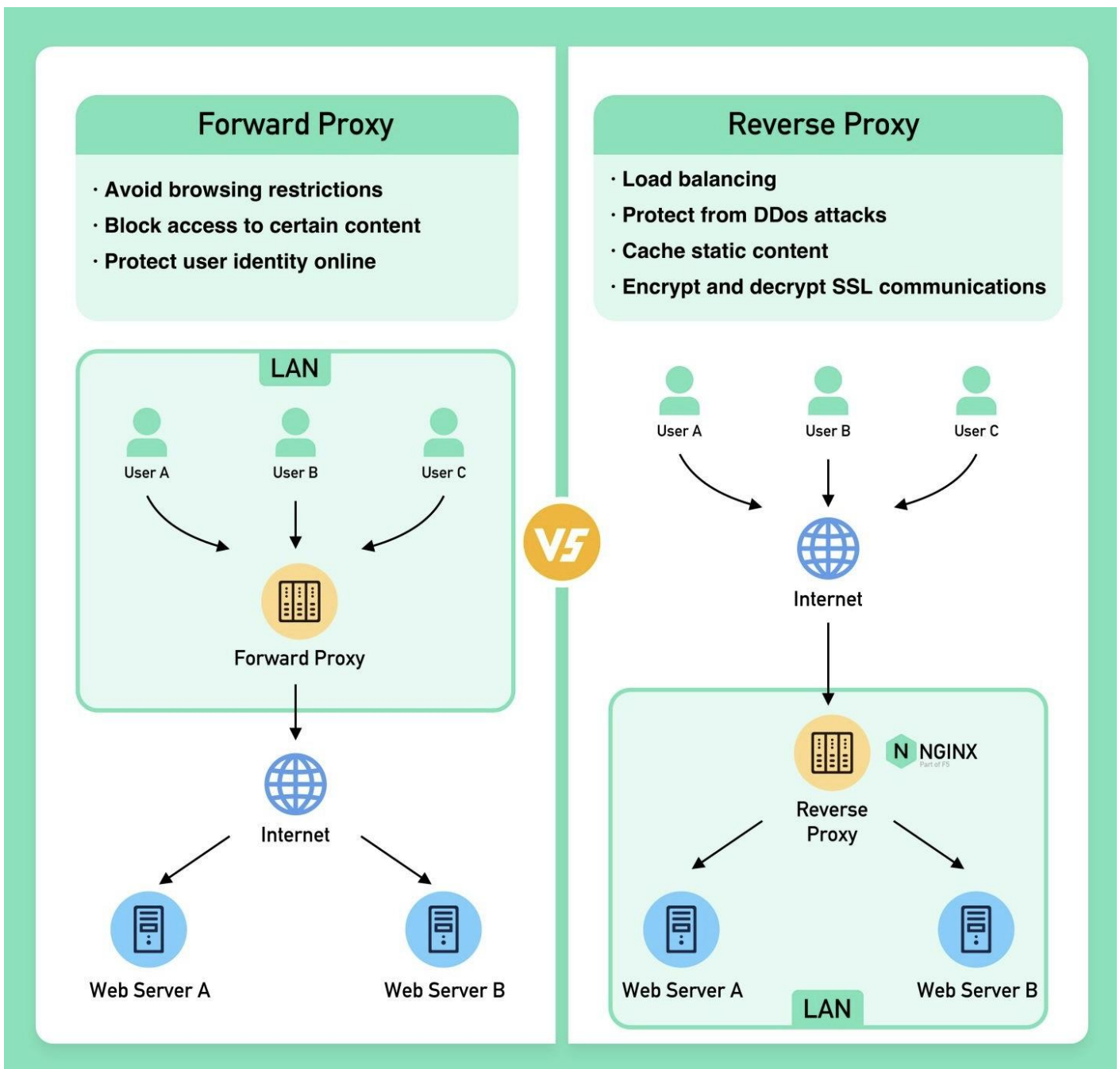


Reverse proxy avec HA proxy

C'est quoi un reverse proxy ?



À l'inverse d'un le reverse proxy se place entre les flux qui sont sur internet et le serveur.

Imaginons un réseau hébergeant un site web accessible sur internet grâce à une redirection de port. Ce site est protégé par un certificat SSL, il utilise donc le port 443 en https.

Si nous souhaitons rendre accessible un deuxième site web en https du même réseau, il nous est impossible de faire deux redirections pour le même port. Nous allons voir qu'avec un reverse proxy c'est tout à fait possible.

Pré-requis

- Vous devez avoir un nom de domaine
- Avoir également un certificat SSL (que ce soit let's encrypt ou autre) voici un tuto pour une wildcard ACME pfsense [ici](#)
- Un site web fonctionnel tournant sur le port 443

Aller dans le gestionnaire de paquet pfsense et installer le paquet HAproxy :

The screenshot shows the pfSense Community Edition web interface. The top navigation bar includes the pfSense logo and the text 'COMMUNITY EDITION'. The 'System' menu is highlighted with a red circle, and its dropdown menu is open, showing options like 'Advanced', 'Certificates', 'General Setup', 'High Availability', 'Package Manager' (highlighted with a red circle), 'Register', and 'Routing'. The 'Interfaces' menu is also visible. The main content area shows the 'System Info' section with fields for 'Name', 'User', and 'System'. Below this, the 'Package Manager' section is active, showing 'Available Packages' (highlighted with a red circle) and a search bar. The search term 'haproxy' is entered in the search bar (highlighted with a red circle). The search results table shows the following data:

Name	Version	Description
haproxy	1.8.15	HAProxy is a free and open source software reverse proxy and load balancer. It can be used to balance traffic between multiple web servers, or to act as a reverse proxy for other applications.

Une fois le paquet installé, vous le trouverez sur la partie [Services](#)

Services ▾

VPN ▾

Stat

Acme Certificates

Auto Config Backup

Captive Portal

DHCP Relay

DHCP Server

DHCPv6 Relay

DHCPv6 Server

DNS Forwarder

DNS Resolver

Dynamic DNS

HAProxy

IGMP Proxy

NTP

PPPoE Server

Router Advertisement

SNMP

Netga

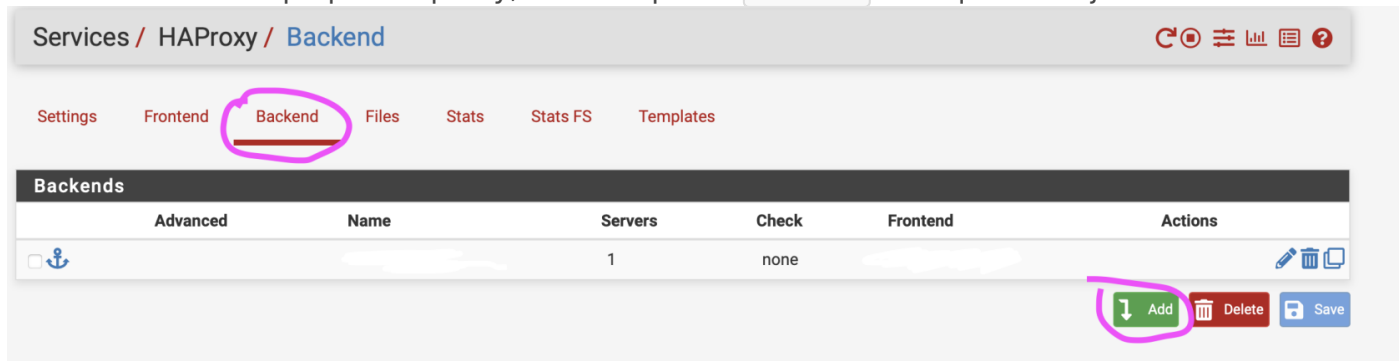
If you p
Comm
hardwar
the **NET**

You also
Support
committ
more th

• **Upg**

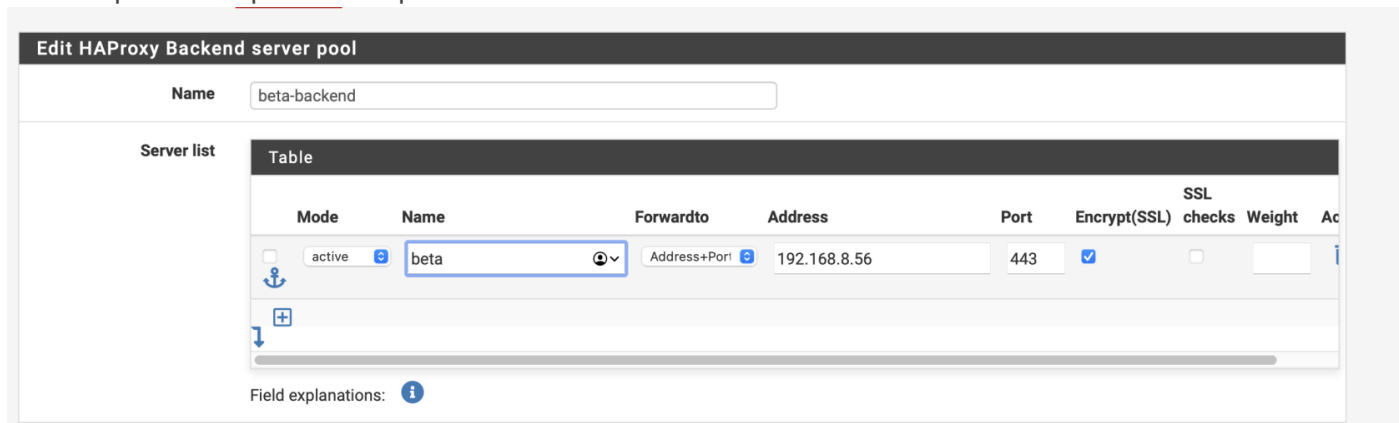
Net

Rendez vous sur le paquet HAProxy, et sur la partie **backend** et cliquer sur ajouter



Le backend sera le service sur lequel on souhaitera redirigé notre trafic

Pour la première partie remplissez là comme ceci



Nous allons appeler notre backend **beta-backend** modifier le nom à votre convenance. Au niveau de la section **Server list** rajoutez une règle, avec le même nom que le backend et le site (il est mieux d'uniformiser les noms pour ne pas se perdre, dans ce tuto nous faisons comme si le site web était beta.mondomaine.com)

Au niveau de Forwardto choisissez Adress+Port, placez y l'adresse ip de la machine hébergeant le site et le port, et cocher le **Encrypt(SSL)**.

On va maintenant aller sur la partie **Frontend** et en créer un nouveau.

Appelez le http-frontend et remplissez le exactement comme ci dessous:

Edit HAProxy Frontend

Name

https-frontend

Description

Status

Active

Shared Frontend

☐ This can be used to host a second or more website on the same IP:Port combination.

Use this setting to configure multiple backends/accesslists for a single frontend.

All settings of which only 1 can exist will be hidden.

The frontend settings will be merged into 1 set of frontend configuration.

External address

Define what ip:port combinations to listen on for incoming connections.

Table

Listen address	Custom address	Port	SSL Offloading	Advanced	
☐ WAN address (IPv4)		443	☒		

NOTE: You must add a firewall rules permitting access to the listen ports above.

If you want this rule to apply to another IP address than the IP address of the interface chosen above, select it here (you need to define [Virtual IP](#) addresses on the first). Also note that if you are trying to redirect connections on the LAN select the "any" option. In the port to listen to, if you want to specify multiple ports, separate them with a comma (,). EXAMPLE: 80,8000 Or to listen on both 80 and 443 create 2 rows in the table where for the 443 you would likely want to check the SSL-offloading checkbox.

Max connections

Sets the maximum amount of connections this frontend will accept, may be left empty.

Type

http / https(offloading)

This defines the processing type of HAProxy, and will determine the available options for acl checks and also several other options. Please note that for https encryption/decryption on HAProxy with a certificate the processing type needs to be set to "http".

Ensuite au niveau des ACL rajouter une nouvelle règle s'appellant **beta**, au niveau des expressions mettez **Host matches** et sur **value**, placez l'url exact de votre site.

Default backend, access control lists and actions

Access Control lists

Use these to define criteria that will be used with actions defined below to perform them only when certain conditions are met.

Table

Name	Expression	CS	Not	Value	Actions
☐ beta	Host matches:	☐	☐	beta.mondomaine.com	

- 'CS' makes the string matches 'Case Sensitive' so www.domain.tld wil not be the same as WWW.domain.TLD

- 'Not' makes the match if the value given is not matched

Example:

Name	Expression	CS	Not	Value
Backend1acl	Host matches			www.yourdomain.tld
addHeaderAc	SSL Client certificate valid			

acl's with the same name will be 'combined' using OR criteria.

For more information about ACL's please see [HAProxy Documentation](#) Section 7 - Using ACL's

NOTE Important change in behaviour, since package version 0.32

-acl's are no longer combined with logical AND operators, list multiple acl's below where needed.

-acl's alone no longer implicitly generate use_backend configuration. Add 'actions' below to accomplish this behaviour.

Maintenant on va faire matcher notre backend qu'on a créer avec notre ACL.
Rajouter une action et sur `use backend` choisissez le beta-backend qu'on a crée et sur `condition acl names` écrivez `beta` qui est le nom de l'ACL qu'on a crée juste avant.
Au niveau de Default Backend mettez none.

-acl's alone no longer implicitly generate use_backend configuration. Add 'actions' below to accomplish this behaviour.

Actions

Use these to select the backend to use or perform other actions like calling a lua script, blocking certain requests or others available.

Table			
Action	Parameters	Condition acl names	Actions
Use Backend	See below	beta	
backend: beta-backend			

Example:

Action	Parameters	Condition
Use Backend	Website1Backend	Backend1acl
http-request header set	Headername: X-HEADER-ClientCertValid New logformat value: YES	addHeaderAcl

Default Backend

None

If a backend is selected with actions above or in other shared frontends, no default is needed and this can be left to "None"

Ici laissez tout comme sur l'image :

Stats options

Separate sockets

☐ Enable collecting & providing separate statistics for each socket.

Logging options

Don't log null

☐ A connection on which no data has been transferred will not be logged.
To skip logging probes from monitoring systems that otherwise would pollute the logging. (It is generally recommended not to use this option in uncontrolled environments (eg: internet), otherwise scans and other malicious activities would not be logged.)

Don't log normal

☐ Don't log connections in which no anomalies are found.
Setting this option ensures that normal connections, those which experience no error, no timeout, no retry nor redispatch, will not be logged.

Raise level for errors

☐ Change the level from "info" to "err" for potentially interesting information.
This option makes haproxy raise the level of logs containing potentially interesting information such as errors, timeouts, retries, redispatches, or HTTP status codes 5xx.

Detailed logging

☐ If checked provides more detailed logging.
Each log line turns into a much richer format including, but not limited to, the connection timers, the session status, the connections numbers, the frontend, backend and server name, and of course the source address and ports. In http mode also the HTTP request and captured headers and cookies will be logged.

Error files

Error files

Use these to replace the error pages that haproxy can generate by custom pages created on the files tab. For example haproxy will generate a 503 error page when no backend is available, you can replace that page here.

Table		
errorcode(s)	Error Page	Actions

Sur Advanced setting laissez les paramètres comme ci dessous :

Advanced settings

Client timeout

the time (in milliseconds) we accept to wait for data from the client, or for the client to accept data (default 30000).

Use "forwardfor" option

☒ Use "forwardfor" option.

The "forwardfor" option creates an HTTP "X-Forwarded-For" header which contains the client's IP address. This is useful to let the final web server know what the client address was. (eg for statistics on domains)

Use "httpclose" option

http-keep-alive (default)

By default HAProxy operates in keep-alive mode with regards to persistent connections: for each connection it processes each request and response, and leaves the connection idle on both sides between the end of a response and the start of a new request.

Bind pass thru

NOTE: paste text into this box that you would like to pass behind each bind option.

Advanced pass thru

NOTE: paste text into this box that you would like to pass thru in the frontend.

Sur la partie SSL Offloading sur **Certificate** choisissez votre wildcard et cochez les deux cases d'en bas comme sur l'image.

SSL Offloading

Note

SSL Offloading will reduce web servers load by maintaining and encrypting connection with users on internet while sending and retrieving data without encryption to internal servers. Also more ACL rules and http logging may be configured when this option is used. Certificates can be imported into the pfSense "Certificate Authority Manager" Please be aware this possibly will not work with all web applications. Some applications will require setting the SSL checkbox on the backend server configurations so the connection to the webserver will also be a encrypted connection, in that case there will be a slight overall performance loss."

SNI Filter

Specify a SNI filter to apply below SSL settings to specific domain(s), see the "crt-list" option from haproxy for details.
EXAMPLE: *.securedomain.tld !public.secdomain.tld

Certificate

wildcard- [redacted]

Choose the cert to use on this frontend.

☒ Add ACL for certificate CommonName. (host header matches the "CN" of the certificate)

☒ Add ACL for certificate Subject Alternative Names.

OCSP

☐ Load certificate ocsp responses for easy certificate validation by the client.

A cron job wil update the ocsp response every hour.

Additional certificates

Which of these certificate will be send will be determined by haproxys SNI recognition. If the browser does not send SNI this will not work properly. (IE on XP is one example, possibly also older browsers or mobile devices).

Table

Certificates	Actions
[redacted]	

☐ Add ACL for certificate CommonName. (host header matches the "CN" of the certificate)

☐ Add ACL for certificate Subject Alternative Names.

Vous pouvez laisser tous le reste par défaut et sauvegarder.

Et là vous pourrez accéder à votre site web sans aucune difficulté via l'url.

Si vous voulez rajouter d'autre sites, il faudra juste créer de nouveau backend et garder le même frontend en rajoutant juste de nouvelles règles au niveau de l'access control list et faisant matcher des actions et ces nouvelles règles.

