ALOHA LOAD BALANCER
MISE EN ŒUVRE DU SSL FRONTEND

« APPNOTES » #0021 — MISE EN ŒUVRE DU SSL FRONTEND

Cette note applicative a pour vocation de vous aider à implémenter la gestion du SSL sur le frontend (connexion vers un serveur HTTP) au sein de la solution ALOHA Load Balancer.

CONTRAINTE

Les utilisateurs extérieurs émettent une requête sécurisée (HTTPS) et les utilisateurs internes continuent de travailler en clair.

OBJECTIF

Implémenter le SSL dans la solution Aloha afin que les utilisateurs qui viennent se connecter sur vos serveurs Web, le fassent grâce à une connexion sécurisée via SSL.

COMPLEXITE

CHANGELOG

2013-01-02: Mise à jour pour ALOHA 5.5 et supérieur

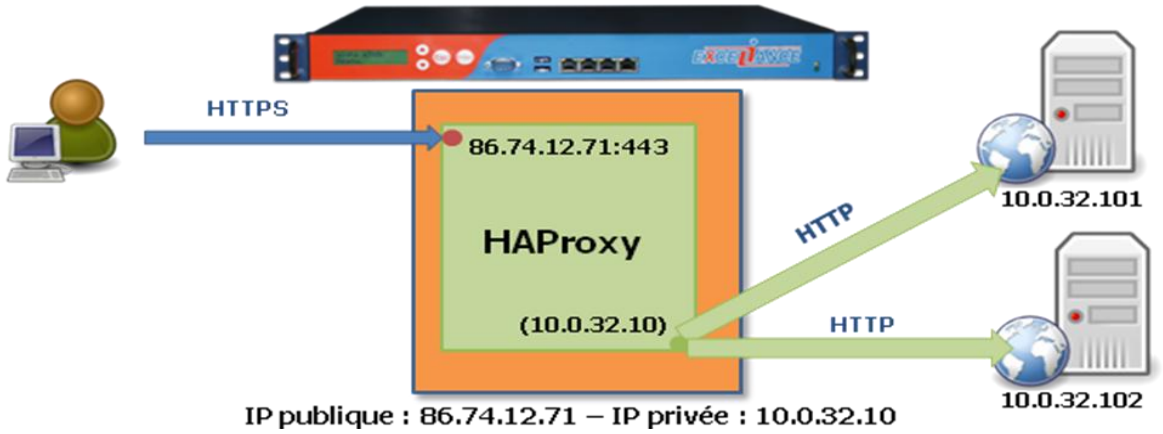
2011-10-21: Mise à jour pour ALOHA 3.7 à 5.0

2010-03-30: initial version

ALOHA 5.5 ET SUPERIEUR

Le composant **stunnel** a disparu depuis l'ALOHA 5.5. Les fonctions SSL sont maintenant gérées directement par le composant **HAProxy**.

SCHEMA CIBLE



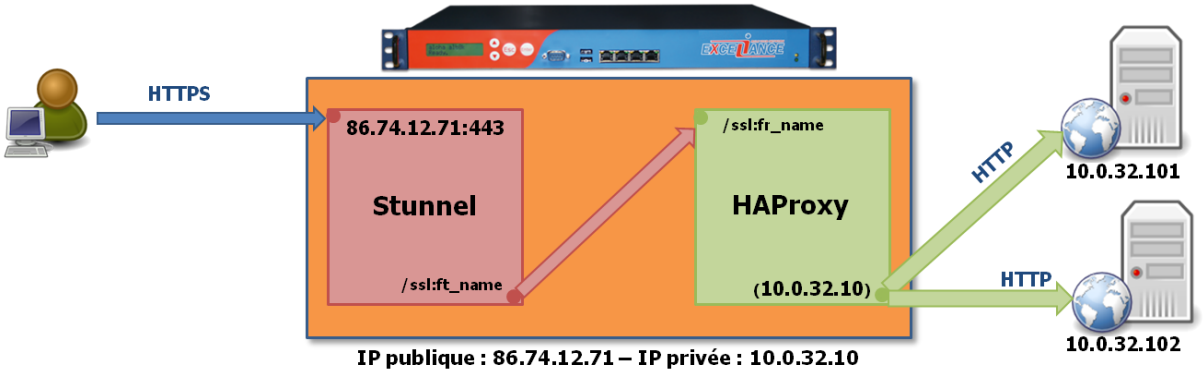
EXTRAIT DE LA CONFIGURATION SSL ET LB NIVEAU 7

Dans l'exemple ci-dessous, domain.com est le nom du certificat dans l'onglet **SSL**.

```
######## The first public address as seen by the clients
frontend frt
 bind  10.0.32.10:80  name http_prv  # address:port to listen to
 bind  86.74.12.71:80 name http_pub  # address:port to listen to
 bind  86.74.12.71:443 name https ssl crt domain.com  # address:port to listen to
 mode http
 log global                          # use global log parameters
 option httplog                      # Enable HTTP logging
 maxconn 4000                        # max conn per instance
 timeout client 25s                  # maximum client idle time
 default_backend bck                 # send everything to this backend by default

####### This backend manages the servers and the load balancing algorithm
backend bck
 balance roundrobin                  # roundrobin | source | uri | leastconn
 mode http
 log global                          # use global log parameters
 option httplog                      # Enable HTTP logging
 cookie  SERVERID insert indirect nocache      # provide persistence with cookie
 option  httpchk HEAD /                        # how to check those servers
 option forwardfor except 127.0.0.1/8          # add X-Forwarded-For except local
 fullconn 4000                                 # dynamic limiting below
 timeout server 25s                            # max server's response time
 server srv1 10.0.32.101:80 cookie s1 weight 10 maxconn 100 check inter 1000 fall 3
 server srv2 10.0.32.102:80 cookie s2 weight 10 maxconn 100 check inter 1000 fall 3
```

SCHEMA CIBLE



EXTRAIT DE LA CONFIGURATION SSL

La configuration de **Stunnel** est accessible directement dans l'onglet **SSL**.

```
; Service-level configuration for frontend
; forward clear requests to haproxy on 127.1.0.x
; and add the xforwarded-for header.
[ssl_frontend]
client = no
key =  /etc/ssl/frontends/SSLfrontend/key.pem
cert = /etc/ssl/frontends/SSLfrontend/crt.pem
accept  = 86.74.12.71:443
connect = /ssl:ft_name
xforwardedfor = yes
```

Vous n'avez besoin de préciser que quelques paramètres lors de l'implémentation d'un proxy SSL :

- `client = no` : place stunnel en mode server
- `key / cert` : chemin d'accès à la clef et au certificat de ce proxy SSL
- `accept` : adresse IP externe où stunnel attend les connections clients
- `connect` : socket unix interne où le traffic est envoyé en clair

EXTRAIT DE LA CONFIGURATION LB NIVEAU7

Après modification de la configuration de **Stunnel** et l'implémentation du(des) certificat(s), il ne reste plus qu'à modifier celle du niveau 7, accessible directement dans l'onglet **LB niveau7**.

Il convient d'ajouter le chemin vers la socket unix d'écoute d'HAProxy qui devra être identique aux paramètres **connect** définis dans la configuration SSL avec le mot clé **accept-proxy**.

```
######## The first public address as seen by the clients
frontend frt
 bind  10.0.32.10:80                 # address:port to listen to
 bind  /ssl:ft_name accept-proxy     # unix socket to listen to
 mode http
 log global                          # use global log parameters
 option httplog                      # Enable HTTP logging
 maxconn 4000                        # max conn per instance
 timeout client 25s                  # maximum client idle time (ms)
 default_backend bck                 # send everything to this backend by default

####### This backend manages the servers and the load balancing algorithm
backend bck
 balance roundrobin                  # roundrobin | source | uri | leastconn
 mode http
 log global                         # use global log parameters
 option httplog                     # Enable HTTP logging
 cookie  SERVERID insert indirect nocache      # provide persistence with cookie
 option  httpchk HEAD /                        # how to check those servers
 option forwardfor except 127.0.0.1/8          # add X-Forwarded-For except local
 fullconn 4000                                 # dynamic limiting below
 timeout server 25s                            # max server's response time
 server srv1 10.0.32.101:80 cookie s1 weight 10 maxconn 100 check inter 1000 fall 3
 server srv2 10.0.32.102:80 cookie s2 weight 10 maxconn 100 check inter 1000 fall 3
```

DEMARRAGE DU SERVICE STUNNEL

IMPORTANT

En cas de première configuration du SSL, un message d'avertissement indique que le service **Stunnel** n'est pas démarré. Dans l'onglet **Service**, éditez la configuration du service **Stunnel** en cliquant sur le bouton «stunnel options».

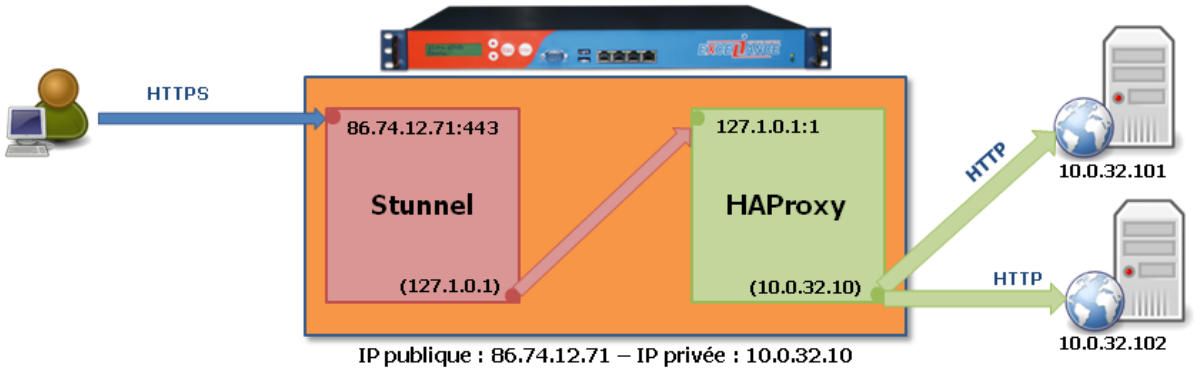
Il suffit de commander la ligne **no autostart** :

```
service stunnel
    ############## The SSL tunnel Daemon
    # no autostart
```

Il ne reste plus qu'à démarrez le service en cliquant sur le bouton **démarrer**.

JUSQU'A ALOHA 3.6

SCHEMA CIBLE



EXTRAIT DE LA CONFIGURATION SSL

La configuration de **Stunnel** est accessible directement dans l'onglet **SSL**.

```
; Service-level configuration for frontend
; forward clear requests to haproxy on 127.1.0.x
; and add the xforwarded-for header.
[ssl_frontend]
client = no
key =  /etc/ssl/frontends/SSLfrontend/key.pem
cert = /etc/ssl/frontends/SSLfrontend/crt.pem
accept  = 86.74.12.71:443
connect = 127.1.0.1:1
xforwardedfor = yes
```

Lors d'une implémentation du SSL en mode frontend, seuls quelques paramètres sont à renseigner :

- le mode de fonctionnement : client ou non SSL (dans le cas présent, le module Stunnel ne devra pas être configuré en mode client mais en mode serveur. L'option «client = no» devra être choisie),
- les chemins de la clé et du certificat créés à l'aide de l'assistant (cf: howto-0020-Mise-en-oeuvre-du-SSL-0912-fr.pdf),
- l'adresse et le port d'écoute liés à un certificat SSL,
- l'adresse et le port de redirection des requêtes à destination d'HAProxy.

EXTRAIT DE LA CONFIGURATION LB NIVEAU7

Après modification de la configuration de Stunnel et l'implémentation du(des) certificat(s), il ne reste plus qu'à modifier celle du niveau 7, accessible directement dans l'onglet **LB niveau7**.

```
######## The first public address as seen by the clients
frontend frt
 bind  10.0.32.10:80              # address:port to listen to
 bind  127.1.0.1:1                # address:port to listen to
 mode http
 log global                       # use global log parameters
 option httplog                   # Enable HTTP logging
 maxconn 4000                     # max conn per instance
 timeout client 25s               # maximum client idle time (ms)
 default_backend bck              # send everything to this backend by default

####### This backend manages the servers and the load balancing algorithm
backend bck
 balance roundrobin               # roundrobin | source | uri | leastconn
 mode http
 log global                       # use global log parameters
 option httplog                   # Enable HTTP logging
 cookie  SERVERID insert indirect nocache    # provide persistence with cookie
 option  httpchk HEAD /                      # how to check those servers
 option forwardfor except 127.0.0.1/8        # add X-Forwarded-For except local
 fullconn 4000                               # dynamic limiting below
 timeout server 25s                          # max server's response time (ms)
 server srv1 10.0.32.101:80 cookie s1 weight 10 maxconn 100 check inter 1000 fall 3
 server srv2 10.0.32.102:80 cookie s2 weight 10 maxconn 100 check inter 1000 fall 3
```

Il convient d'ajouter alors l'adresse et le port d'écoute d'HAProxy qui devront être identiques aux paramètres «connect» définis dans la configuration du SSL.

DEMARRAGE DU SERVICE STUNNEL

IMPORTANT

En cas de première configuration du SSL, un message d'avertissement indique que le service «Stunnel» n'est pas démarré. Dans l'onglet Service, éditez la configuration du service Stunnel en cliquant sur le bouton «stunnel options».

```
service stunnel
    ############## The SSL tunnel Daemon
    # config <dir>      : daemon configuration file
    config /etc/stunnel/stunnel.conf
    # no autostart                  # commenter le no devant autostart
```

Il ne reste plus qu'à démarrez le service en cliquant sur le bouton «démarrer».