ALOHA LOAD BALANCER
MANAGING SSL ON THE FRONTEND

"APPNOTE" #0021 — MANAGING SSL ON THE FRONTEND

This application note is intended to help you implement SSL management on the frontend (connection to an HTTP server) within the ALOHA Load Balancer solution.

CONSTRAINT

External users submit a secure request (HTTPS) and internal users continue to work over an unencrypted connection.

PURPOSE

Implement SSL in the Aloha solution so that users connect to your Web servers via a secure SSL connection.

COMPLEXITY

CHANGELOG

2013-01-02: update for ALOHA 5.5 and above

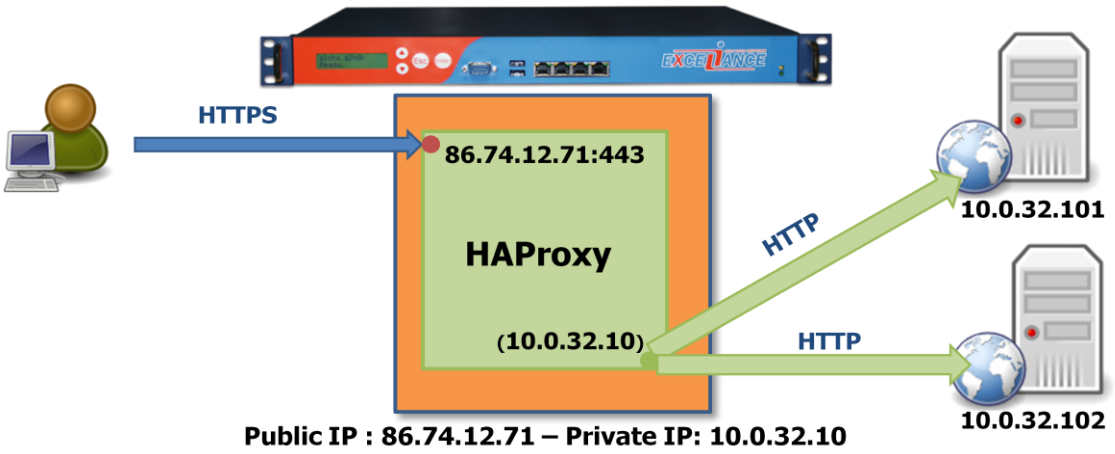
2011-10-21: update for ALOHA 3.7 to 5.0

2010-03-30: initial version

ALOHA 5.5 AND ABOVE

Stunnel component has disappeared in ALOHA 5.5. SSL features are now handled directly by **HAProxy** component.

TARGET NETWORK DIAGRAM



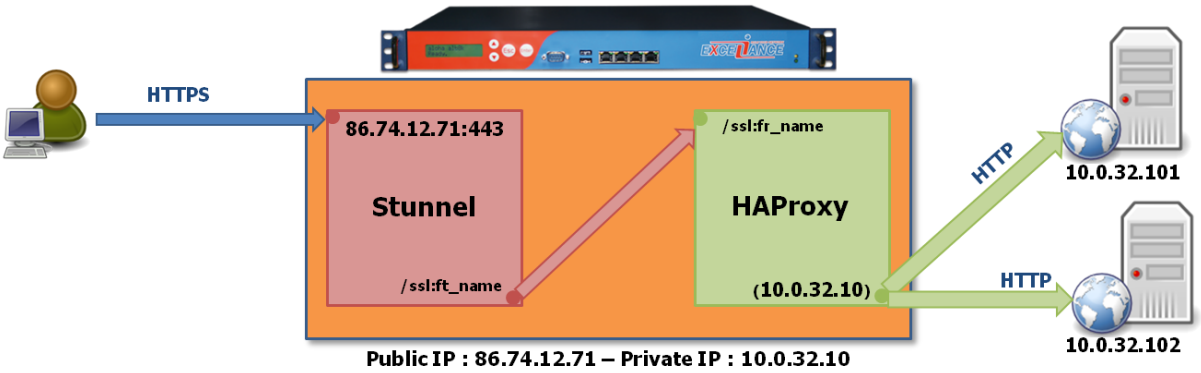
LB LAYER 7 AND SSL CONFIGURATION EXAMPLE

In the example below, domain.com is the SSL certificate name from the **SSL** tab.

```
######## 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
```

TARGET NETWORK DIAGRAM



EXTRACT OF THE SSL CONFIGURATION

You can directly access the **Stunnel** configuration from the **SSL** tab.

```
; 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
```

You only need to specify a few parameters when implementing a SSL frontend:

- `client = no` : tells stunnel to work in server mode
- `key / cert` : path to the key and the certificate for this SSL frontend
- `accept` : external IP address where stunnel expect (encrypted) client connections
- `connect` : internal unix socket where to forward unencrypted traffic

EXTRACT OF THE LB LEVEL7 CONFIGURATION

After generating or installing the SSL certificate and modifying the **Stunnel** configuration you need to modify the LB level7 configuration. Just click on the **LB level7** tab.

Add the HAProxy unix sockt binding. It must be identical to the one specified in the **connect** parameters defined in the **SSL** configuration and add the keyword **accept-proxy** at the end.

```
######## 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 (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
```

STARTING THE STUNNEL SERVICE

IMPORTANT

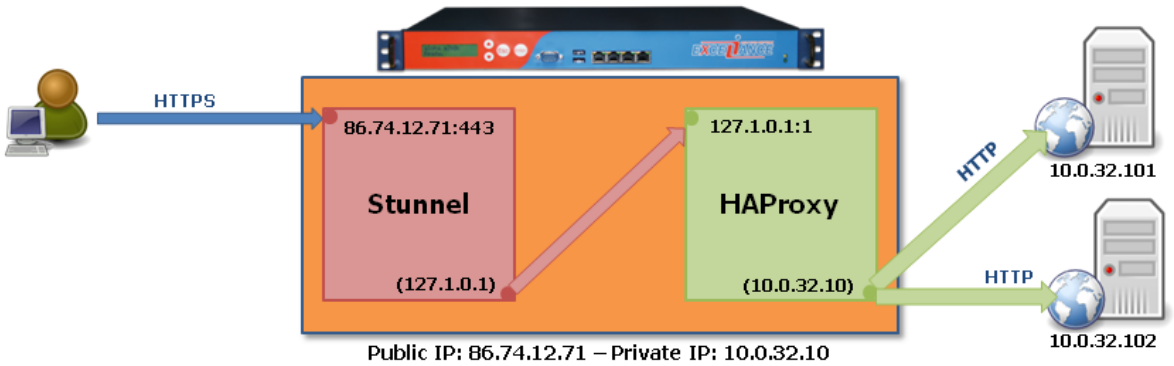
When you first configure SSL, a warning message indicates that the **Stunnel** service has not started. In the **Service** tab, edit the **Stunnel service** configuration by clicking on the stunnel options button. Just comment out the line **no autostart**:

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

Now you simply need to start the service by clicking the **start** button.

ALOHA 3.6 AND BELOW

TARGET NETWORK DIAGRAM



EXTRACT OF THE SSL CONFIGURATION

You can directly access the **Stunnel** configuration from the **SSL** tab.

```
; 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
```

You only need to specify a few parameters when implementing SSL in frontend mode:

- the operating mode: client or non-SSL (in this case, the Stunnel module should not be configured in client mode, but in server mode. Therefore you should choose the "client = no" option)
- the paths for the key and the certificate created using the wizard (see howto-0020-Implementing-SSL-0912-fr.pdf)
- the address and listening port linked to an SSL certificate
- the address and redirection port for requests to HAProxy

EXTRACT OF THE LB LEVEL7 CONFIGURATION

```
######## 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
```

After modifying the Stunnel configuration and the implementation(s) of the certificate(s), all that remains is to modify the configuration of level 7. You can do this from the LB level7 tab.

Add the HAProxy address and listening port; they must be identical to those specified in the " connect" parameters defined in the SSL configuration.

STARTING THE STUNNEL SERVICE

IMPORTANT

When you first configure SSL, a warning message indicates that the "Stunnel" service has not started. In the Service tab, edit the configuration of the Stunnel service by clicking the "stunnel options" button.

```
service stunnel
    ############## The SSL tunnel Daemon
    # config <dir>      : daemon configuration file
    config /etc/stunnel/stunnel.conf
    # no autostart # add a comment no before autostart
```

Now you simply need to start the service by clicking the "start" button.