HA Proxy

* High availability proxy
* It is a TCP/HTTP load balancer and proxy server that allows a webserver to spread incoming requests across multiple endpoints

ACL

* Access Control List (ACL) is a test condition
* Actions are performed based on the result of the test conditions
* Syntax

acl <aclname> <criterion> [flags] [operator] [<value>]

acl acl\_myApp path\_sub myApp

a**cl\_myapp**: The name of the ACL.

**path\_sub: A function that validates if the request URL has myApp as a substring.**

**BACKEND**

* This is a set of servers that actually processes the forwarded requests. The backend consists of a load balancing algorithm and a list of servers with ports.
* Syntax

backend <backendname>

balance <loadbalancing algorithm>

server <name of the server> <ip>:<port> check

....

server <name of the server> <ip>:<port> check

* Ex

backend myAppBackend

balance roundrobin

server myAppServer1 172.21.28.1:8080 check

server myAppServer2 172.21.28.2:8080 check

* myAppBackend: The name of the backend to which the ACL will forward the request.
* roundrobin: The name of the load balancing algorithm.
* myAppServer, myAppServer1: The name of the server.
* 172.21.28.1, 172.21.28.2: IP addresses of servers.
* 8080: The port on which server listens.
* check: Specifies to check the health of the server.

**FrontEnd**

* The frontend defines how requests are forwarded to backends
* It is composed of set of IP addresses, a port, ACLs, and use\_backend rules.
* Syntax

frontend <frontend name>

bind <IPs or wild card>:80

acl <aclname> <criterion> [flags] [operator] [<value>] ...

use\_backend <backend name> if <aclname>

* Ex.

frontend myAppFrontEnd

bind \*:80

acl acl\_myApp path\_sub myApp

use\_backend myAppBackend if acl\_myApp

**myAppFrontEnd**: Name of the frontend.

The frontend listens on *port 80* to all the interfaces \*  available in server. If the URL has *myApp* as a substring, the request is forwarded to myAppBackend. Otherwise, you will get no service available exception.

**Statistics (Optional)**

The statistics report of HAProxy will show the status of the servers, the number of connections, etc

his can be enabled easily by adding the following configuration to the config file:

syntax

listen stats

bind \*:<Port>

stats enable

stats hide-version

stats uri </url>

stats auth <username>:<password>

ex.

listen stats

bind \*:9999

stats enable

stats hide-version

stats uri /stats

stats auth admin:admin@123

**9999**: Listens on *port 9999* for statistics report requests.

**/stats**: The URL of the statistics page (i.e.*localhost:9999/stats*).

**admin**/**admin@123**: The username and password to access the statistics page.