Contents

[1 Introduction 4](#_Toc521327997)

[1.1 Purpose 4](#_Toc521327998)

[1.2 Scope 4](#_Toc521327999)

[1.3 References 4](#_Toc521328000)

[2 Load Balancer high level description 4](#_Toc521328001)

[2.1 HA-Proxy 5](#_Toc521328002)

[2.2 KeepAlived 9](#_Toc521328003)

# Document Revision History

| revision | Date | Updated By | Description |
| --- | --- | --- | --- |
| PA1 | 2018-06-26 | Juan José García | Initial Draft |
| PA2 | 2018-08-06 | Ivan Martinez Navas | Review |
| PA3 | 2018-09-20 | Ivan Martinez Navas | Add notify keeAlived |
|  |  |  |  |

# Introduction

## Purpose

This manual is a guide to install and configure IAP-AMX Load Balancer.

## Scope

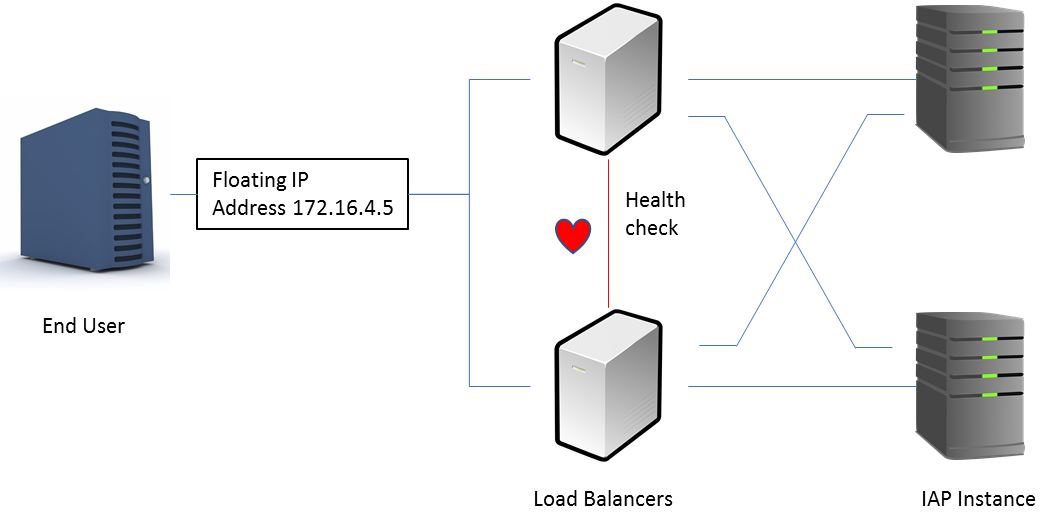
This manual is addressed to the Ericsson technical staff.

## References

# Load Balancer high level description

A load balancer improves the distribution of workloads across multiple computer resources. Load balancing aims sits between the end users and the servers. Not only does the load balancer distribute the requests between all the backend servers, but also checks its health to avoid any request is sent to a not available backend.

To provide complete high availability in the system, the load balancer function needs also redundancy: two load balancers running in active/passive cluster configuration in which each load balancer monitors the other and share a Virtual IP. The active node will be handing the VIP and receiving all the IP requests. In the event that the active node fails, the other node is promoted to active and assigned the VIP without service affectation.



Our load balance solution is composed by following components:

* HA-Proxy. Is a load balancer software. It listen for incoming connections in the VIP and forwards to the server pool. HA-Proxy continuously monitors the server nodes to avoid forwarding requests to not accessible servers.
* Keepalive: this is a simple cluster software. It decides which node should be the master and assigns the VIP to it. In case that the active node fails, keepalive will promote the passive node to active state and will assign the VIP. For a node to be promoted as active, both Keepalived and HA-Proxy needs to be working on the node.

To avoid brain split situations (both nodes think they are the active node) the cluster interface must be same network interface where the service is provided (and the VIP assigned). If the two nodes don’t have connectivity between them, then any other element in the network will necessarily only be able to connect to one of the nodes.

## HA-Proxy

HA-Proxy load balancer is a standard Red Hat 7 component. Installation and configuration is described in Red Hat documentation. This section only provides the recommended configuration for IAP. This needs to be done as root user in all required nodes.

### Install HA-Proxy

To install HA-Proxy following these instructions:

# yum install haproxy

### Configure HA-Proxy load balancer.

Edit or create **/etc/haproxy/haproxy.cfg** file and add following content

#---------------------------------------------------------------------

# Example configuration for a possible web application. See the

# full configuration options online.

#

# http://haproxy.1wt.eu/download/1.4/doc/configuration.txt

#

#---------------------------------------------------------------------

#---------------------------------------------------------------------

# Global settings

#---------------------------------------------------------------------

global

# to have these messages end up in /var/log/haproxy.log you will

# need to:

#

# 1) configure syslog to accept network log events. This is done

# by adding the '-r' option to the SYSLOGD\_OPTIONS in

# /etc/sysconfig/syslog

#

# 2) configure local2 events to go to the /var/log/haproxy.log

# file. A line like the following can be added to

# /etc/sysconfig/syslog

#

# local2.\* /var/log/haproxy.log

#

log 127.0.0.1 local2

#log loghost local0 info

chroot /var/lib/haproxy

pidfile /var/run/haproxy.pid

maxconn 4096

user haproxy

group haproxy

# turn on stats unix socket

stats socket /var/lib/haproxy/stats

#---------------------------------------------------------------------

# common defaults that all the 'listen' and 'backend' sections will

# use if not designated in their block

#---------------------------------------------------------------------

defaults

mode http

log global

option httplog

option dontlognull

option http-server-close

option forwardfor except 127.0.0.0/8

option redispatch

retries 3

timeout http-request 10s

timeout queue 1m

timeout connect 10s

timeout client 1m

timeout server 1m

timeout http-keep-alive 10s

timeout check 10s

maxconn 3000

#---------------------------------------------------------------------

# SSL frontend which proxys to the backends

#---------------------------------------------------------------------

frontend https-in \*:443

mode tcp server iapdev01 172.16.4.13:8443 check

use\_backend iap-https

#---------------------------------------------------------------------

# round robin balancing between the various backends

#---------------------------------------------------------------------

backend iap-https

balance roundrobin

mode tcp

# maximum SSL session ID length is 32 bytes.

stick-table type binary len 32 size 30k expire 30m

# Use the real server names and IPs

server iapdev01 172.16.4.14:8443 check

server iapdev02 172.16.4.15:8443 check

server iapdev03 172.16.4.16:8443 check

server iapdev04 172.16.4.17:8443 check

#---------------------------------------------------------------------

# HTTP redirecto to SSL port

#---------------------------------------------------------------------

frontend http-in \*:80

redirect scheme https code 301

### Configure kernel parameters

To allow haproxy to bind to the VIP address (even when it is not assigned) add following line

net.ipv4.ip\_nonlocal\_bind=1

To **/etc/sysctl.conf** system configuration file and force kernel parameters reload:

# sysctl -p

### Configure HA-Proxy autostart.

Enable ha-proxy to be started at boot time:

# systemctl enable haproxy.service

And enable to be re-started automatically whenever it crashes or is killed:add following under *[service]*section in **/etc/systemd/system/multi-user.target.wants/haproxy.service** file

Restart=always

Finally reload the service:

# systemctl daemon-reload

# systemctl restart haproxy.service

### Configure firewall to enable HA-Proxy traffic.

Enable firewall to receive incoming connections in TCP port 443 (https). Create or edit **/etc/firewalld/services/haproxy-https.xml** file with the content indicated below:

<?xml version="1.0" encoding="utf-8"?>

<service>

<short>HAProxy-HTTP</short>

<description>HAProxy load-balancer</description>

<port protocol="tcp" port="443"/>

</service>

Reapply SELINUX security directives

# restorecon /etc/firewalld/services/haproxy-https.xml

# chmod 640 /etc/firewalld/services/haproxy-https.xml

Reload firewall:

# firewall-cmd –reload

### HA-Proxy logging

Configure rsyslog daemon to log HA-Proxy in its own file. Create or edit **/etc/rsyslog.d/haproxy.conf** with following content

local2.\* /var/log/haproxy.log

If required that logs go to a different machine, then edit **/etc/rsyslog.conf** file to enable the UDP port 514 to be used by ha-proxy and uncomment following lines

$ModLoad imudp  
$UDPServerRun 514

If you want to use a specific IP, you can add a new line like the real IP

$UDPServerAddress 127.0.0.1

Save and restart rsyslog

systemctl restart rsyslog

## KeepAlived

KeepAlived cluster software is a standard Red Hat 7 component. Installation and configuration is described in Red Hat documentation. This section only provides the recommended configuration for IAP. This needs to be done as root user in all required nodes.

### Install KeepAlived

To install KeepAlived just execute following command: instructions:

# yum install keepalived

# yum install psmisc # for killall

### Configure KeepAlived

Edit or create **/etc/keepalived.conf** with following content.

# Script used to check if HAProxy is running

vrrp\_script chek\_haproxy { # Requires keepalived-1.1.13

script "killall -0 haproxy" # cheaper than pidof

interval 2 # check every 2 seconds

weight 2 # add 2 points of prio if OK

}

# Virtual interface

# The priority specifies the order in which the assigned interface to take over in a failover

vrrp\_instance VIP {

interface eth0

state **ROLE** # replace ROLE with MASTER or SLAVE

priority **XX** # 99 on master, 100 on backup

virtual\_router\_id 33

# The virtual ip address shared between the two loadbalancers

virtual\_ipaddress {

**172.16.5.A**

}

unicast\_src\_ip **172.16.5.B** -- this node private IP

unicast\_peer {

**172.16.5.C** -- other nodes in cluster private IPs

}

authentication {

auth\_type PASS

auth\_pass load-balancer-password

}

track\_script {

chek\_haproxy

}

notify /var/iap/bin/notify-keepalived.sh

}

Finally it is necessary to customize previous script for each node, as following:

* Replace state *ROLE*with **MASTER** or **SLAVE** depending on the desired initial state. Note only one node should be configured to start in master state.
* Set a value for priority, according to the selected state role. Replace *XX* with **99** for MASTER, replace *XX* with **100** for SLAVE. If more than one slave, decrease priority in one for each additional slave.
* Update the *virtual\_ipaddress*, *unicast\_src\_ip* and *peer\_ip* with the correct IPs. Also add more *peer\_ip* elements if more nodes are added.
* To improve security, modify the password token in *auth\_pass*.

### Configure KeepAlived for automatic start

Enable KeepAlived to be started at boot time:

# systemctl enable keepalived.service

And enable to be re-started automatically whenever it crashes or is killed: add following under *[service]*section in **/etc/systemd/system/multi-user.target.wants/keepalived.service** file

Restart=always

Finally reload the service:

# systemctl daemon-reload

# systemctl restart keepalived.service

### KeepAlived logging

We need to configure rsyslog daemon to log the KeepAlived logs. First edit **/etc/sysconfig/keepalived** and edit or add following line to instruct KeepAlived to log to rsyslog *local6* facility:

KEEPALIVED\_OPTIONS="-D --log-facility=6"

Configure rsyslog daemon to log HA-Proxy in its own file. Create of edit **/etc/rsyslog.d/keepalived.conf** with following content

Local6.\* /var/log/keepalived.log

If required that logs go to a different machine, then edit **/etc/rsyslog.conf** file to enable the UDP port 514 to be used by ha-proxy and uncomment following lines

$ModLoad imudp  
$UDPServerRun 514

If you want to use a specific IP, you can add a new line like the real IP:

$UDPServerAddress 127.0.0.1

Save and restart rsyslog

systemctl restart rsyslog

### KeepAlived notify

Add in contrab KeepAlived KPI

Install these scripts:

* /var/iap/bin/notify-keepalived.sh
* /var/iap/bin/AMXKPI00.sh