

## Monitor cluster heartbeat lines IP reahability via ping ICMP protocol with Zabbix

Author : admin



Say you're having an haproxy load balancer cluster with two or more nodes and you are running the servers inside some complex **organizational hybrid complex network** that is a combination of a local **DMZ lans**, many switches, dedicated connectivity lines and every now and then it happens for the network to mysteriously go down. Usually simply setting monitoring on the **network devices CISCO** itself or the smart switches used is *enough to give you an overview on what's going on but if haproxy is in the middle of the end application servers and in front of other Load balancers* and network equipment sometimes it might happen that due to failure of a **network equipment / routing issues or other strange unexpected reasons one of the 2 nodes connectivity might fail down via the configured dedicated additional Heartbeat lines that are usually configured in order to keep away the haproxy CRM Resource Manager cluster** thus ending it up in a split brain scenarios.

Assuming that this is the case like it is with us you would definitely want to keep an eye on the connectivity of **Connect Line1** and **Connect Line2** inside some monitoring software like zabbix. As our company main monitoring software used to monitor our infrastructure is *Zabbix* in this little article, I'll briefly explain **how to configure the network connectivity status change from haproxy node1 and haproxy node2 Load balancer cluster to be monitored via a simple ICMP ping echo checks**.

Of course the easies way to configure an ICMP monitor via Zabbix is using **EnableRemoteCommands=1** inside `/etc/zabbix/zabbix-agentd.conf` but if your infrastructure should be of High Security and PCI perhaps this options is prohibited to be used on the servers. This is why to

achieve still the ICMP ping checks with **EnableRemoteCommands=0** a separate simple bash user parameter script could be used. **Read further to find out one way ICMP monitoring with a useparameter script can be achieved with Zabbix.**

### 1. Create the userparameter check for heartbeat lines

```
root@haproxy1 zabbix_agentd.d]# cat userparameter_check_heartbeat_lines.conf
UserParameter=heartbeat.check,\
/etc/zabbix/scripts/check_heartbeat_lines.sh
```

```
root@haproxy2 zabbix_agentd.d]# cat userparameter_check_heartbeat_lines.conf
UserParameter=heartbeat.check,\
/etc/zabbix/scripts/check_heartbeat_lines.sh
```

### 2. Create check\_heartbeat\_lines.sh script which will be actually checking connectivity with simple ping

```
root@haproxy1 zabbix_agentd.d]# cat /etc/zabbix/scripts/check_heartbeat_lines.sh
#!/bin/bash
hb1=haproxy2-lb1
hb2=haproxy2-lb2
if ping -c 1 $hb1 &> /dev/null
then
    echo "$hb1 1"
else
    echo "$hb1 0"
fi
if ping -c 1 $hb2 &> /dev/null
then
    echo "$hb2 1"
```

```
else
    echo "$hb2 0"
fi
[root@haproxy1 zabbix_agentd.d]#
```

```
root@haproxy2 zabbix_agentd.d]# cat /etc/zabbix/scripts/check_heartbeat_lines.sh
#!/bin/bash
hb1=haproxy1-hb1
hb2=haproxy1-hb2
if ping -c 1 $hb1 &> /dev/null
then
    echo "$hb1 1"
else
    echo "$hb1 0"
fi
if ping -c 1 $hb2 &> /dev/null
then
    echo "$hb2 1"
else
    echo "$hb2 0"
fi
[root@haproxy2 zabbix_agentd.d]#
```

### 3. Test script heartbeat lines first time

Each of the nodes from the cluster are properly pingable via ICMP protocol

The script has to be run on both haproxy1 and haproxy2 Cluster (load) balancer nodes

```
[root@haproxy-hb1 zabbix_agentd.d]# /etc/zabbix/scripts/check_heartbeat_lines.sh
haproxy2-hb1 1
haproxy2-hb2 1
```

```
[root@haproxy-hb2 zabbix_agentd.d]# /etc/zabbix/scripts/check_heartbeat_lines.sh
haproxy1-hb1 1
haproxy1-hb2 1
```

The status of 1 returned by the script should be considered remote defined haproxy node is reachable / 0 means ping command does not return any ICMP status pings back.

#### **4. Restart the zabbix-agent on both cluster node machines that will be conducting the ICMP ping check**

```
[root@haproxy zabbix_agentd.d]# systemctl restart zabbix-agentd
[root@haproxy zabbix_agentd.d]# systemctl status zabbix-agentd
...
[root@haproxy zabbix_agentd.d]# tail -n 100 /var/log/zabbix_agentd.log
...
```

#### **5. Create Item to process the userparam script**

Create Item as follows:

Item
Preprocessing

Parent items
Pacemaker

\* Name
Check heartbeat lines

Type
Zabbix agent (active)

\* Key
heartbeat.check

Type of information
Character

\* Update interval
5m

Custom intervals

Type	Interval	Period	Action
Flexible	Scheduling	50s	1-7,00:00-24:00
Add Remove			

\* History storage period
Do not keep history
Storage period
90d

Show value
As is

New application

Applications

Logserver
Maintenance
Memory
Network interfaces
OS
Pacemaker

Populates host inventory field
-None-

Description
The item is intended to check the heartbeat lines.

Enabled
☒

Update
Clone
Execute now
Test
Clear history and trends
Delete
Cancel

## 6. Create the Dependent Item required

Item Preprocessing

Parent items Pacemaker

\* Name check heartbeat line 2

Type Dependent item

\* Key heartbeat2

\* Master item afipve100776: Check heartbeat lines Select

Type of information Character

\* History storage period Do not keep history Storage period 90d i

Show value As is

New application

Applications

Logserver

Maintenance

Memory

Network interfaces

OS

Pacemaker

Populates host inventory field -None-

Description

Enabled ☒

Update

Clone

Execute now

Test

Clear history and trends

Delete

Cancel

Item Preprocessing

Preprocessing steps

Name

Parameters

Custom on fail

Actions

1: Regular expression

hb1(s+)(d+)

v2

☐

Test Remove

Add

Test all steps

Update

Clone

Execute now

Test

Clear history and trends

Delete

Cancel

For processing you need to put the following simple regular expression

Name: **Regular Expression**

Parameters: **hb1(\s+)(\d+)**

Custom on fail: **\2**

Item
Preprocessing

Preprocessing steps

	Name	Parameters	Custom on fail	Actions
1	Regular expression	hb2(\s+)(\d+)	\2	<input type="checkbox"/> <a href="#">Test</a> <a href="#">Remove</a>

Add

Update
Clone
Execute now
Test
Clear history and trends
Delete
Cancel

[Test all steps](#)

<input type="checkbox"/>	WARNING	OK	P03 Pacemaker: <span style="background-color: yellow;">beat</span> beat line 1 on (HOSTNAME) is DOWN	{	<span style="background-color: red;">fqdn1</span>	<span style="background-color: red;">last()</span> <1	Enabled
<input type="checkbox"/>	WARNING	OK	P03 Pacemaker: <span style="background-color: yellow;">beat</span> beat line 2 on (HOSTNAME) is DOWN	{	<span style="background-color: red;">fqdn2</span>	<span style="background-color: red;">last()</span> <1	Enabled

## 7. Create triggers that will be generating the Alert

Create the required triggers as well

Trigger
Tags
Dependencies

Parent triggers
Pacemaker

\* Name
heartbeat line 2 on {HOST.NAME} is DOWN

Operational data

Severity
Not classified
NORMAL
WARNING
MINOR
MAJOR
CRITIC

\* Expression
{FQDN:heartbeat2.last()}<1
Add

Expression constructor

OK event generation
Expression
Recovery expression
None

PROBLEM event generation mode
Single
Multiple

OK event closes
All problems
All problems if tag values match

Allow manual close
☐

URL

Description

Enabled
☒

Update
Clone
Delete
Cancel

Main thing to configure here in Zabbix is below expression

Expression: {FQDN:heartbeat2.last()}