Motadata – HA

Configuration Guide

Table of Contents

Copyright Notice

Notices

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT CAREFULLY BEFORE DOWNLOADING OR USING THE SOFTWARE. BY CLICKING ON THE "I ACCEPT THE TERMS OF THE LICENSE AGREEMENT" BUTTON, OPENING THE PACKAGE, DOWNLOADING THE PRODUCT, OR USING THIS PRODUCT, YOU ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE WITH ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE "I DO NOT ACCEPT THE TERMS OF THE LICENSE AGREEMENT" BUTTON AND THE INSTALLATION PROCESS WILL NOT CONTINUE. RETURN THE PRODUCT TO THE PLACE OF PURCHASE FOR A FULL REFUND, OR DO NOT DOWNLOAD THE PRODUCT. YOUR GENERAL TERMS OF BUSINESS DO NOT APPLY.

General
In this software license agreement:
a) "Mindarray" means Mindarray Systems Pvt Ltd., 14/3, Magnet Corporate Park. 100 Feet Road, S.G Highway, Near Sola Bridge Opp. Grand Cambay, Thaltej, Ahmedabad, Gujarat. India.
b) "Customer" means the individual(s), organization or business entity buying a license of the software from Mindarray or its distributors or its resellers.
c) "Software" means computer programs (and their storage medium) supplied by Mindarray and known collectively as "Mindarray IP Address Manager" in which Mindarray has propriety rights for its any user manuals, example code, operating instructions, brochures and all other documentation relating to the said computer programs (the expression "software" to include all or any part or any combination of software).

License Grant
This license grants you the following rights:

a) Software product: Mindarray grants to you an exclusive license to use the software for the sole purposes of designing, developing, and testing your software components or applications ("applications"). You may install the software on any computer in your organization.
b) Electronic documents: Solely with respect to electronic documents included with the software, you may make an unlimited number of copies (either in hardcopy or electronic form), provided that such copies shall be used only for internal purposes and are not republished or distributed to any third party.
c) License file: A file provided at the time of sale uniquely identifies each license. This license grant is contingent upon the purchase of a license file from Mindarray or one of Mindarray's resellers.
d) Sample code: Mindarray grants you the right to use and modify the source code parts of the software that are listed in the "projects" and "scripts" subdirectories (if available).
e) Redistribution: The software is made available for download solely for use by end users according to the license agreement. Any reproduction or redistribution of the software not in accordance with the license agreement is expressly prohibited.
f) Trial software: If the software is installed without a serial number then, notwithstanding other sections of this license, you may use the software for up to 30 days after installation.
g) Not for resale software: If the software is labeled as "not for resale " or "NFR" then, notwithstanding other sections of this license, you may not resell, or otherwise transfer the value of software, neither distribute any redistributables.
h) Reservation of rights: Mindarray reserves all rights not expressly granted to you in this license agreement. The license is granted to the customer on a non-exclusive-basis which means that Mindarray will grant the license also to their individuals, organizations and business entities.
i) This license agreement consists of no obligations for Mindarray to offer support (services), help (services) or maintenance (services) relating to the software. Obligations for Mindarray to offer maintenance (services) relating to the software can only arise from a maintenance agreement between Mindarray and customer. General terms of business of the customer do not apply.

Upgrades and Supplements
If the software is labeled as an upgrade, you must be properly licensed to use a product identified by Mindarray as being eligible for the upgrade in order to use the software. Software labeled as an upgrade replaces and/or supplements the product that formed the basis for your eligibility for the upgrade. You may use the resulting upgraded product only in accordance with the terms of this license unless we provide other terms along with the update or supplement. If the software is an upgrade of a component or a package or software programs that you licensed as a single product, the software may be used and transferred only as part of that single product package.

Limitation on Reverse Engineering, Decompilation, And Disassembly
Customer may not reverse engineer, decompile, or disassemble the software, except and only to the extent that it is expressly permitted by applicable law notwithstanding this limitation.

Termination
Without prejudice to any other rights, Mindarray may cancel or dissolve this license agreement if the customer does not abide by the terms and conditions of this license agreement, in which case customer must destroy all copies of the software and all of its component parts.

Limited Warranty
Mindarray warrants that for a period of ninety (90) days from the date of shipment from Mindarray. The media on which the software is furnished will be free of defects in materials and workmanship under normal use.

The software substantially conforms to its published specifications. Except for the foregoing, the software is provided as is. This limited warranty extends only to the customer as the original licensee. Customer's exclusive remedy and the entire liability of Mindarray and its suppliers under this limited warranty will be, at Mindarray or its service center's option, repair, replacement, or refund of the software if reported (or, upon request, returned) to the party supplying the software to the customer. In no event does Mindarray warrants that the software is error-free or that customer will be able to operate the software without problems or interruptions. The customer will safeguard Mindarray against any claim relating to the use of the software by the customer. This warranty does not apply if the software: (a) has been altered, except by Mindarray; (b) has not been installed, operated, repaired, or maintained in accordance with instructions supplied by Mindarray; (c) has been subjected to abnormal physical or electrical stress misuse, negligence, or accident; (d) is used in high-risk activities, including the operation of nuclear facilities, aircraft navigation, air traffic control, weapons systems, life support or medical applications for use in any circumstance in which the failure of the software could lead directly to death, personal injury or damage to properties or the environment.

Limitation of Liability and Remedies

NOTWITHSTANDING ANY DAMAGES THAT YOU MIGHT INCUR FOR ANY REASON WHATSOEVER (INCLUDING, WITHOUT LIMITATION ALL INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OR MULTIPLE DAMAGES SUCH AS BUT NOT LIMITED TO, LOST BUSINESS OR PROFITS, LOSS OF GOODWILL, WORKS TOP PAGE AND DATA LOSS), THE ENTIRE LIABILITY OF MINDARRAY AND ANY OF ITS SUPPLIERS UNDER ANY PROVISION OF THIS LICENSE AGREEMENT AND YOUR EXCLUSIVE REMEDY FOR ALL OF THE FOREGOING (EXCEPT FOR ANY REMEDY OF REPAIR OR REPLACEMENT ELECTED BY MINDARRAY WITH RESPECT TO ANY BREACH OF THE LIMITED WARRANTY) SHALL BE LIMITED TO THE AMOUNT ACTUALLY PAID BY YOU FOR THE SOFTWARE. MINDARRAY IS RELIEVED OF ANY OBLIGATION TO PAY DAMAGES IF THE CUSTOMER HAS NOT UPGRADED THE SOFTWARE WHEN POSSIBLE. THE FOREGOING LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS (INCLUDING SECTIONS 4, 5 AND 6 ABOVE) SHALL APPLY TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, EVEN IF ANY REMEDY FAILS ITS ESSENTIAL PURPOSE.

Entire Agreement

This license agreement (including any addendum or amendment to this license agreement which is included with the software) is the entire agreement between you and Mindarray relating to the software and the support services (if any) and they supersede all prior or contemporaneous oral or written communications, proposals and representations with respect to the software or any other subject matter covered by this license agreement. To the extent the terms of any Mindarray policies or programs for support services conflict with the terms of this license agreement, the terms of this license agreement shall control.

The customer is not allowed to alienate or transfer any rights relating to this license agreement without the written approval of Mindarray.

THIS AGREEMENT SHALL BE CONSTRUED IN ACCORDANCE WITH THE LAWS OF THE INDIAN GOVERNMENT AND THE INDIAN COURTS SHALL HAVE SOLE JURISDICTION IN ANY DISPUTE RELATING TO THESE CONDITIONS. ALL DISPUTES HEREUNDER SHALL BE RESOLVED EXCLUSIVELY IN THE APPROPRIATE COURT IN THE CITY OF AHMEDABAD, INDIA. If any part of these conditions shall be or become invalid or unenforceable in any way and to any extent by any existing or future rule of law, order, statute or regulation applicable thereto, then the other conditions shall remain in full force and effect as all other provisions.

The conditions of this license agreement remain applicable after the termination of this license agreement if this results from the nature of the condition.

Copyright

The software is protected by copyright and other intellectual property laws and treaties. Mindarray or its suppliers own the title, copyright, and other intellectual property rights in the software. The granting of a license does not constitute a transfer of any intellectual property right. The software is licensed, not sold.

Pre-requisite

Below are pre-requisite for Motadata HA Functionality.

1. Procure required Server/VM as per deployment scenario.
2. Procure Physical IP and Virtual IP as per deployment scenario.
3. Install Master setup and Slave setup using Motadata Installation guide.
4. Update Master setup and Slave setup License as per requirement.
5. Make sure Slave server has all plugin available in Plugin and Plugin-lib folder, otherwise copy it from Master.
6. Configure email notification to get notified in adversities.
7. Generate key in observer and put in all servers for root login.

Open Points

- Seamless License sync between Master and Slave.

1. Introduction

1.1 About HA

Motadata network monitoring system (NMS) can be configured in master-slave High Availability modes for uninterrupted services. With this feature, your Network is always monitored by NMS even in unforeseen scenarios and system Failures. The whole HA setup is managed through physical connectivity to prevent any data loss or communication failure.

In master-slave mode, all your network is monitored through one NMS server (Master server, master RPE and master Report DB). The master server also updates the slave NMS server (Slave server, slave RPE and slave Report DB) in real time. The slave NMS is always in standby mode for any unfortunate scenario. When the master NMS goes down, the slave NMS takes in charge immediately.

The observer unit monitors the connectivity of master and slave servers. The Observer decides who and when should be treated as a master. When the master server doesn't respond, observer changes the slave server into master server. Clickhouse service needs to be kept in up status for HA to work, or else HA sync will not work properly.

1.2 Terminology

Master Server: The Config DB and GUI components are stored in same server. The Config DB stores the configuration data. Based on its own and RPE inputs, Master server updates the master report DB and slave report DB. The server keeps an updated copy of its data into the slave server.

Slave Server: It is the clone of master server. The server is constantly updated by master server. When master goes down, observer unit makes this as master server.

Master RPE: The RPE unit facilitates faster monitor polling. It takes monitor Information from Config DB and polls them. It forwards the information back into Master server.

Slave RPE: It is the clone of master RPE. When master RPE fails, the slave RPE Works as master RPE.

Master Report DB: Report DB is the main reporting server. It talks to Master server for information sharing.

Slave Report DB: It is the clone of master report DB. It is used in case of failure.

Observer Utility: It monitors the connectivity of servers and takes decision based on it. When master server is down it configures the slave server to act as master.

NSQ Channel: NSQ channels are created on master server for all replication hosts.

2. Configuration of Keepalived

- Please follow below mention steps to install and configure Keepalived in master and slave server

1. curl --progress http://keepalived.org/software/keepalived-1.2.15.tar.gz | tar xz
2. cd keepalived-1.2.15
3. ./configure
4. make
5. sudo make install
6. ln -s /usr/local/sbin/keepalived /usr/sbin/keepalived
7. check by issuing "keepalived --version"
8. vim /etc/init.d/keepalived -> copy from file in folder
9. chmod +x /etc/init.d/keepalived
10. update-rc.d keepalived defaults
11. #cp /usr/local/etc/keepalived/keepalived.conf /etc/keepalived/keepalived.conf
12. mkdir /etc/keepalived/
13. vim /etc/keepalived/keepalived.conf -> copy from file in folder
14. /etc/init.d/keepalived start or "service keepalived start" and "service keepalived stop"

- Please use below configuration for keepalived.conf

```
! Configuration File for Master keepalived

vrrp_instance VI_1 {
   state BACKUP
   nopreempt
   interface ens32
   virtual_router_id 51
   priority 100
   advert_int 1
   authentication {
      auth_type PASS
      auth_pass motadata
   }
   virtual_ipaddress {
      172.16.10.51
   }
}

! Configuration File for Slave keepalived

vrrp_instance VI_1 {
   state BACKUP
   nopreempt
   interface ens32
```

```
    virtual_router_id 51
    priority 50
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass motadata
    }
    virtual_ipaddress {
        172.16.10.51
    }
}
```

- Please use below configuration for keepalived

```
#! /bin/sh
#
# keepalived   LVS cluster monitor daemon.
#
#             Written by Andres Salomon <dilinger@voxel.net>
#
### BEGIN INIT INFO
# Provides:          keepalived
# Required-Start:    $syslog $network
# Required-Stop:     $syslog $network
# Default-Start:     2 3 4 5
# Default-Stop:      1
# Short-Description: Starts keepalived
# Description:       Starts keepalived lvs loadbalancer
### END INIT INFO
PATH=/sbin:/bin:/usr/sbin:/usr/bin
DAEMON=/usr/sbin/keepalived
NAME=keepalived
DESC=keepalived
CONFIG=/etc/keepalived/keepalived.conf
TMPFILES="/tmp/.vrrp /tmp/.healthcheckers"

#includes lsb functions
. /lib/lsb/init-functions

set -e
test -f $CONFIG || exit 0
test -f $DAEMON || exit 0

case "$1" in
  start)
        log_daemon_msg "Starting $DESC" "$NAME"
        for file in $TMPFILES
        do
                test -e $file && test ! -L $file && rm $file
        done
        if start-stop-daemon --start --quiet --pidfile /var/run/$NAME.pid \
```

```
            --exec $DAEMON; then
                log_end_msg 0
        else
                log_end_msg 1
        fi
        ;;
  stop)
        log_daemon_msg "Stopping $DESC" "$NAME"
        if start-stop-daemon --oknodo --stop --quiet --pidfile /var/run/$NAME.pid \
                --exec $DAEMON; then
                log_end_msg 0
        else
                log_end_msg 1
        fi
        ;;
  reload|force-reload)
        log_action_begin_msg "Reloading $DESC configuration..."
        if start-stop-daemon --stop --quiet --signal 1 --pidfile \
              /var/run/$NAME.pid --exec $DAEMON; then
              log_end_msg 0
        else
                log_action_end_msg 1
        fi
        ;;
  restart)
        log_action_begin_msg "Restarting $DESC" "$NAME"

        start-stop-daemon --stop --quiet --pidfile \
                /var/run/$NAME.pid --exec $DAEMON
        sleep 1
        if start-stop-daemon --start --quiet --pidfile \
              /var/run/$NAME.pid --exec $DAEMON; then
              log_end_msg 0
        else
                log_end_msg 1
        fi
        ;;
  *)
        echo "Usage: /etc/init.d/$NAME {start|stop|restart|reload|force-reload}" >&2
        exit 1
        ;;
esac

exit 0
```

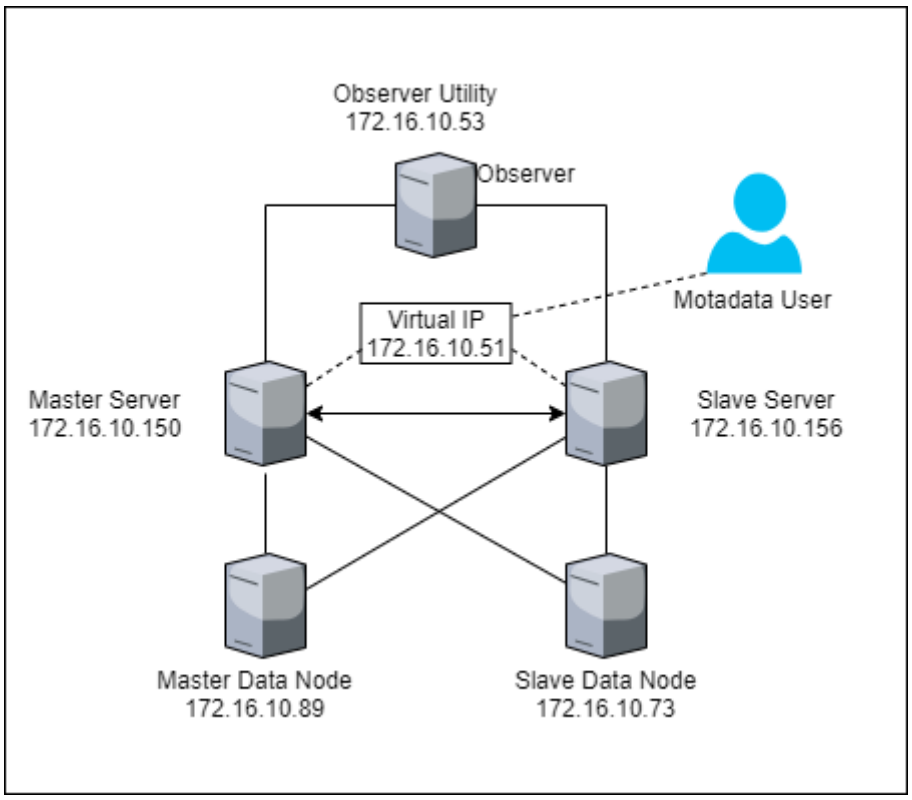
3. Distributed Scenario

3.1 HA Configuration

3.1.1 IP Addresses

- Master Server – 172.16.10.150
- Slave Server – 172.16.10.156
- Master Data Node – 172.16.10.89
- Slave Data Node – 172.16.10.73
- Observer Utility – 172.16.10.53
- Virtual IP – 172.16.10.51

3.1.2 Diagram



3.2 Master – Slave Server Configuration

3.2.1 Master Server Configuration

Please do required configuration in below mention configuration file:

- Cluster-conf.yml file.
- db-engine-conf.yml file.
- nsq-conf.yml file.

3.2.1.1 Cluster-conf.yml file

```
installation-type: 1
target-hosts: 172.16.10.156,172.16.10.73
config-db-host: 172.16.10.150
master-host: 172.16.10.150
current-host: 172.16.10.150
cluster-migration-strict-bind: false
query-backup-duration: 2
```

1. Connect with Master Server – 172.16.10.150.
2. Open /motadata/motadata/config/**cluster-conf.yml** file.
3. Configure installation type to 1.
4. Configure target hosts with slave and slave data Node IP: 172.16.10.156, 172.16.10.73
5. Config-db-host to Master IP: 172.16.10.150
6. Master-host IP: 172.16.10.150
7. Current-host to Master server's IP: 172.16.10.150
8. Save and Exit the file.

3.2.1.2 DB-engine-conf.yml file

1. Open /motadata/motadata/config/**db-engine-conf.yml** file.
2. Configure host IP with Master data node IP: 172.16.10.89.
3. Configuration replication-host with slave data node IP: 172.16.10.73

```
#db engine host
host: 172.16.10.89

#other replicated clickhouse hosts
replication-host: 172.16.10.73

#db engine port
port: 8123

connection-timeout: 60
```

3.2.1.3 Nsq-conf.yml file

1. Open /motadata/motadata/config/nsq-conf.yml file.
2. Configure host with master server's IP: 172.16.10.150.

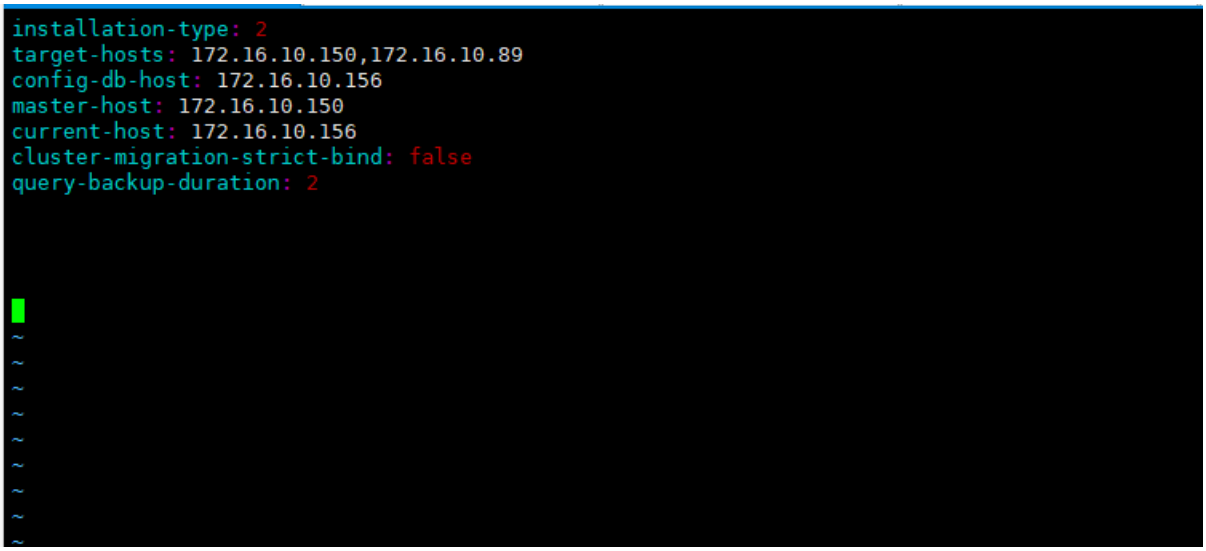
```
# nsq host
host:  172.16.10.150
#nsq port
port: 4150
#nsq lookupd port
lookupd-port: 4160
#nsq lookup port
lookup-port: 4161
#nsq http port
lookup-http-port: 4151
#nsq message in memory
message-queue-size: 20000
#nsq message timeout (seconds)
message-timeout: 600
#nsq message size(bytes)
max-message-size: 200000000
#metric db insert batch size
metric-db-batch-size: 20000
max-message-helper: '4'
cluster-message-queue-size: '0'
ssl-connection: "disabled"
ssl-port: "4152"
```

3.3 Slave Server Configuration

Please do required configuration in below mention configuration file.

- Cluster-conf.yml file.
- db-engine-conf.yml file.
- nsq-conf.yml file.

3.3.1 Cluster-conf.yml



1 Connect with the slave server – 172.16.10.156.
2 Open /motadata/motadata/config/**cluster-conf.yml** file.
3 Configure installation type to **2**
4 Configure target hosts with Master server and Master data node IP: 172.16.10.150,172.16.10.89
5 Configure config-db-host with slave master's IP: 172.16.10.156
6 Configure master-host with master server's IP: 172.16.10.150
7 Configure config-db-host with slave master's IP: 172.16.10.156
8 Save and Exit the file.

3.3.2 DB-engine-conf.yml

1 Open /motadata/motadata/config/db-engine-conf.yml file.
2 Configure host with slave data node IP: 172.16.10.73
3 Configure replication-host with master data node IP: 172.16.10.89

```
#db engine host
host: 172.16.10.73

#other replicated clickhouse hosts
replication-host: 172.16.10.89

#db engine port
port: 8123

connection-timeout: 60
```

3.3.3 Nsq-conf.yml file

1 Open /motadata/motadata/config/**nsq-conf.yml** file.
2 Configure host with slave server's IP: 172.16.10.156.

```
# nsq host
host:  172.16.10.156
#nsq port
port: 4150
#nsq lookupd port
lookupd-port: 4160
#nsq lookup port
lookup-port: 4161
#nsq http port
lookup-http-port: 4151
#nsq message in memory
message-queue-size: 20000
#nsq message timeout (seconds)
message-timeout: 600
#nsq message size(bytes)
max-message-size: 200000000
#metric db insert batch size
metric-db-batch-size: 20000
max-message-helper: '4'
cluster-message-queue-size: '0'
ssl-connection: "disabled"
ssl-port: "4152"
```

3.4 Master – Slave Data Node

3.4.1 Configure Master Data Node

Please do required configuration in below mention configuration file

- Cluster-conf.yml file.

3.4.1.1 Cluster-conf.yml

```
# installation type (0 = stand alone no cluster, 1 = cluster master, 2 = cluster client)
installation-type: 2

# report database ip/host address for (Clickhouse)
report-db-host: 172.16.10.89

# master machine ip/host address
master-host: 172.16.10.150

# current(This) machine ip/host address
current-host: 172.16.10.89

# data backup while failure up to in days (minimum 1 , max 7) (applied on report data only)
query-backup-duration: 2
```

1 Connect with the master data node IP – 172.16.10.89.
2 Open /motadata/motadata/config/**cluster-conf.yml** file.
3 Configure installation type to **2**.
4 Configure report-db-host with master data node IP – 172.16.10.89
5 Configure master-host with master server's IP: 172.16.10.150
6 Configure current-host with master data node IP: 172.16.10.89
7 Save and exit the file.

3.4.2 Configure Slave Data Node

Please do required configuration in below mention configuration file

1. Cluster-conf.yml file.

3.4.2.1 Cluster-conf.yml

```
# installation type (0 = stand alone no cluster, 1 = cluster master, 2 = cluster client)
installation-type: 2

# report database ip/host address for (Clickhouse)
report-db-host: 172.16.10.73

# master machine ip/host address
master-host: 172.16.10.150

# current(This) machine ip/host address
current-host: 172.16.10.73

# data backup while failure up to in days (minimum 1 , max 7) (applied on report data only)
query-backup-duration: 2
```

1 Connect with the slave data node IP – 172.16.10.73.
2 Open /motadata/motadata/config/**cluster-conf.yml** file.
3 Configure installation type to **2**.
4 Configure report-db-host with slave data node IP – 172.16.10.73
5 Configure master-host with master server's IP: 172.16.10.150
6 Configure current-host to slave server's IP: 172.16.10.73
7 Save and Exit the file.

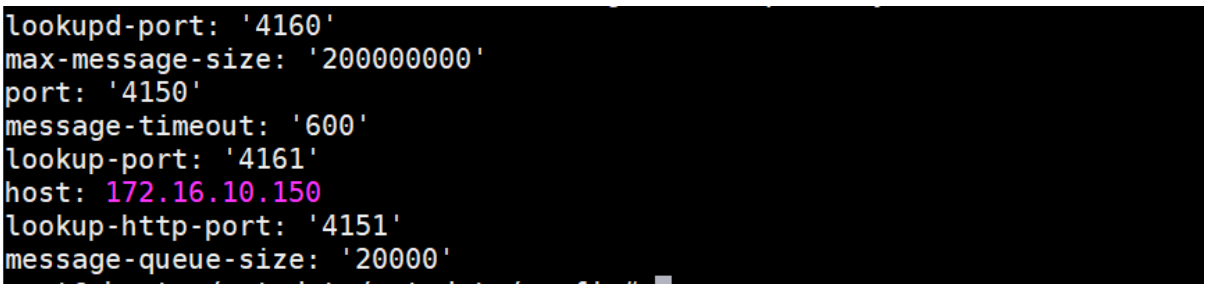
3.5 Configure Observer Utility

3.5.1 Observer-conf.yml file

```
#schedule duration in seconds [check master host connectivity on every provided seconds] (not allowed less then 10 second)
schedule-duration: 10

#all config database hosts
config-db-host: 172.16.10.150,172.16.10.156

#all report database hosts
report-db-host: 172.16.10.89,172.16.10.73

#current system host
current-observer-host: 172.16.10.53

#reconnect time in seconds [consider master as down after this much seconds connectivity not available] (not allowed less then
10 second)
reconnect-time: 60

#currently running or alive as master host
current-master-host: 172.16.10.150:8080

#ssh connection port for all hosts
ssh-port: 22

#master hosts which are not alive as master
target-host: 172.16.10.156:8080

#rpe hosts
rpe-target-host:

#currently down hosts
down-target-host:

#delete logs after days
log-retention-days: 2
```

2. Configure conf-db-host with Master Server IP and Slave Server IPs: 172.16.10.150, 172.16.10.156.
3. Configure report-db-host with master data node IP and slave data node IP: 172.16.10.89, 172.16.10.73.
4. Configure current-observer-host with observers utility's IP: 172.16.10.53.
5. Configure current-master-host with master server IP and port number: 172.16.10.150:8080
6. Configure target-host with slave server IP and port number: 172.16.10.156:8080.

3.5.2 nsq-conf.yml



7. Configure host with master server's IP: 172.16.10.150

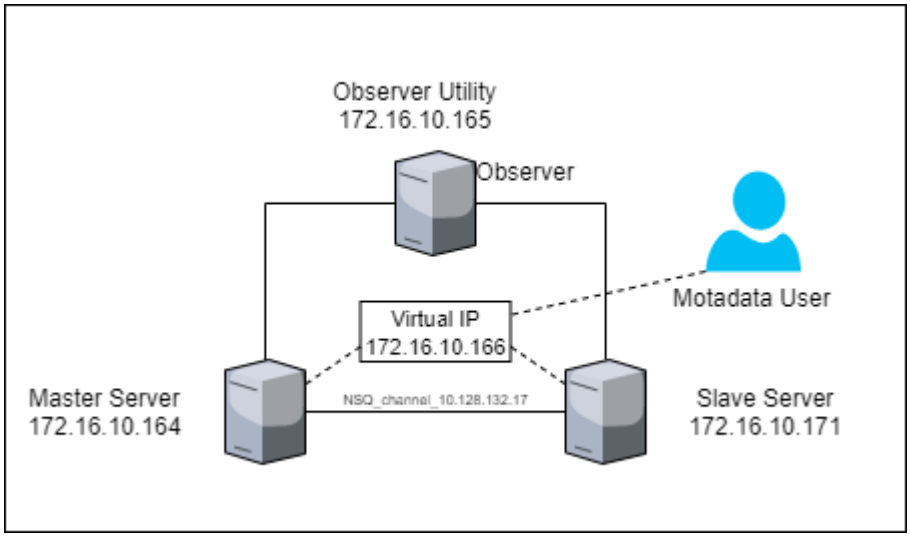
4. Standalone Scenario

4.1 HA Configuration

4.1.1 IP Addresses

- Master Server – 172.16.10.164
- Slave Server – 172.16.10.171
- Observer Utility – 172.16.10.165
- Virtual IP – 172.16.10.166

4.1.2 Diagram



4.2 Master –Server Configuration

4.2.1 Master Server Configuration

Please do required configuration in below mention configuration file:

- Cluster-conf.yml file.
- db-engine-conf.yml file.
- nsq-conf.yml file.

4.2.1.1 Cluster-conf.yml file

```
# installation type (0 = stand alone no cluster, 1 = cluster master, 2 = cluster client)
installation-type: 1

#------------------------------------------------------------- Only for Master Node ----------------------------------------------------------

#required in master server only
target-hosts: 172.16.10.171

#------------------------------------------------------------ Only for Receiver Cluster Node -------------------------------------------------------
# config database ip/host address for (H2)
config-db-host: 172.16.10.164

# report database ip/host address for (click house server for single box HA)
report-db-host: 172.16.10.164

# master machine ip/host address
master-host: 172.16.10.164

# current(This) machine ip/host address
current-host: 172.16.10.164

# data backup while failure up to in days (minimum 1 , max 7) (applied on report data only)
query-backup-duration: 2

cluster-migration-strict-bind": "false"
```

1. Connect with Master Server – 172.16.10.164.
2. Open /motadata/motadata/config/**cluster-conf.yml** file.
3. Configure installation type to 1.
4. Configure target hosts 172.16.10.171
5. Config-db-host: 172.16.10.164
6. report-db-host: 172.16.10.164
7. Master-host IP: 172.16.10.164
8. Current-host: 172.16.10.164
9. Save and Exit the file.

4.2.1.2 DB-engine-conf.yml file

1. Open /motadata/motadata/config/**db-engine-conf.yml** file.
2. Configure host IP: localhost.
3. Configuration replication-host: 172.16.10.171

```
#db engine host
host: localhost

#other replicated clickhouse hosts
replication-host: 172.16.10.171

#db engine port
port: 8123

connection-timeout: 60
```

4.2.1.3 Nsq-conf.yml file

1. Open /motadata/motadata/config/nsq-conf.yml file.
2. Configure host: 172.16.10.164

```
# nsq host
host: 172.16.10.164
#nsq port
port: 4150
#nsq lookupd port
lookupd-port: 4160
#nsq lookup port
lookup-port: 4161
#nsq http port
lookup-http-port: 4151
#nsq message in memory
message-queue-size: 20000
#nsq message timeout (seconds)
message-timeout: 600
#nsq message size(bytes)
max-message-size: 200000000
#metric db insert batch size
metric-db-batch-size: 20000

max-message-helper": "20000"
```

4.3 Slave –Server Configuration

4.3.1 Slave Server Configuration

Please do required configuration in below mention configuration file:

- Cluster-conf.yml file.
- db-engine-conf.yml file.
- nsq-conf.yml file.

4.3.1.1 Cluster-conf.yml file

```
# installation type (0 = stand alone no cluster, 1 = cluster master, 2 = cluster client)
installation-type: 2

#------------------------------------------------------------- Only for Master Node ---------------------------------------------------------

#required in master server only
target-hosts: 172.16.10.164

#------------------------------------------------------- Only for Receiver Cluster Node -----------------------------------------------------
# config database ip/host address for (H2)
config-db-host: 172.16.10.171

# report database ip/host address for (click house server for single box HA)
report-db-host: 172.16.10.171

# master machine ip/host address
master-host: 172.16.10.164

# current(This) machine ip/host address
current-host: 172.16.10.171

# data backup while failure up to in days (minimum 1 , max 7) (applied on report data only)
query-backup-duration: 2

cluster-migration-strict-bind": "false"
```

1. Connect with Slave Server – 172.16.10.171.
2. Open /motadata/motadata/config/**cluster-conf.yml** file.
3. Configure installation type to 2.
4. Configure target hosts 172.16.10.164
5. Config-db-host: 172.16.10.171
6. report-db-host: 172.16.10.171
7. Master-host IP: 172.16.10.164
8. Current-host: 172.16.10.171
9. Save and Exit the file.

4.3.1.2 DB-engine-conf.yml file

1. Open /motadata/motadata/config/**db-engine-conf.yml** file.
2. Configure host IP: localhost.
3. Configuration replication-host: 172.16.0.164

```
#db engine host
host: localhost

#other replicated clickhouse hosts
replication-host: 172.16.10.164

#db engine port
port: 8123

connection-timeout: 60
```

4.3.1.3 Nsq-conf.yml file

1. Open /motadata/motadata/config/nsq-conf.yml file.

2. Configure host: 172.16.10.171

```
# nsq host
host: 172.16.10.171
#nsq port
port: 4150
#nsq lookupd port
lookupd-port: 4160
#nsq lookup port
lookup-port: 4161
#nsq http port
lookup-http-port: 4151
#nsq message in memory
message-queue-size: 20000
#nsq message timeout (seconds)
message-timeout: 600
#nsq message size(bytes)
max-message-size: 200000000
#metric db insert batch size
metric-db-batch-size: 20000

max-message-helper": "20000"
```

4.4 Configure Observer Utility

4.4.1 Observer-conf.yml file

```
1   schedule-duration: '10'
2   config-db-host: 172.16.10.164,172.16.10.171
3   report-db-host: 172.16.10.164,172.16.10.171
4   current-observer-host: 172.16.10.165
5   reconnect-time: '60'
6   current-master-host: 172.16.10.164:8080
7   ssh-port: '22'
8   target-host: 172.16.10.171:8080
9   rpe-target-host: ''
10  down-target-host: ''
11  log-retention-days: '7'
12
```

8. Configure conf-db-host: 172.16.10.164, 172.16.10.171.
9. Configure report-db-host: 172.16.10.164, 172.16.10.171.
10. Configure current-observer-host: 172.16.10.165.
11. Configure current-master-host: 172.16.10.164:8080
12. Configure target-host: 172.16.10.164:8080.

4.4.2 nsq-conf.yml

```
lookupd-port: '4160'
max-message-size: '200000000'
port: '4150'
message-timeout: '600'
lookup-port: '4161'
host: 172.16.10.164
lookup-http-port: '4151'
message-queue-size: '20000'
```

- Configure host: 172.16.10.164

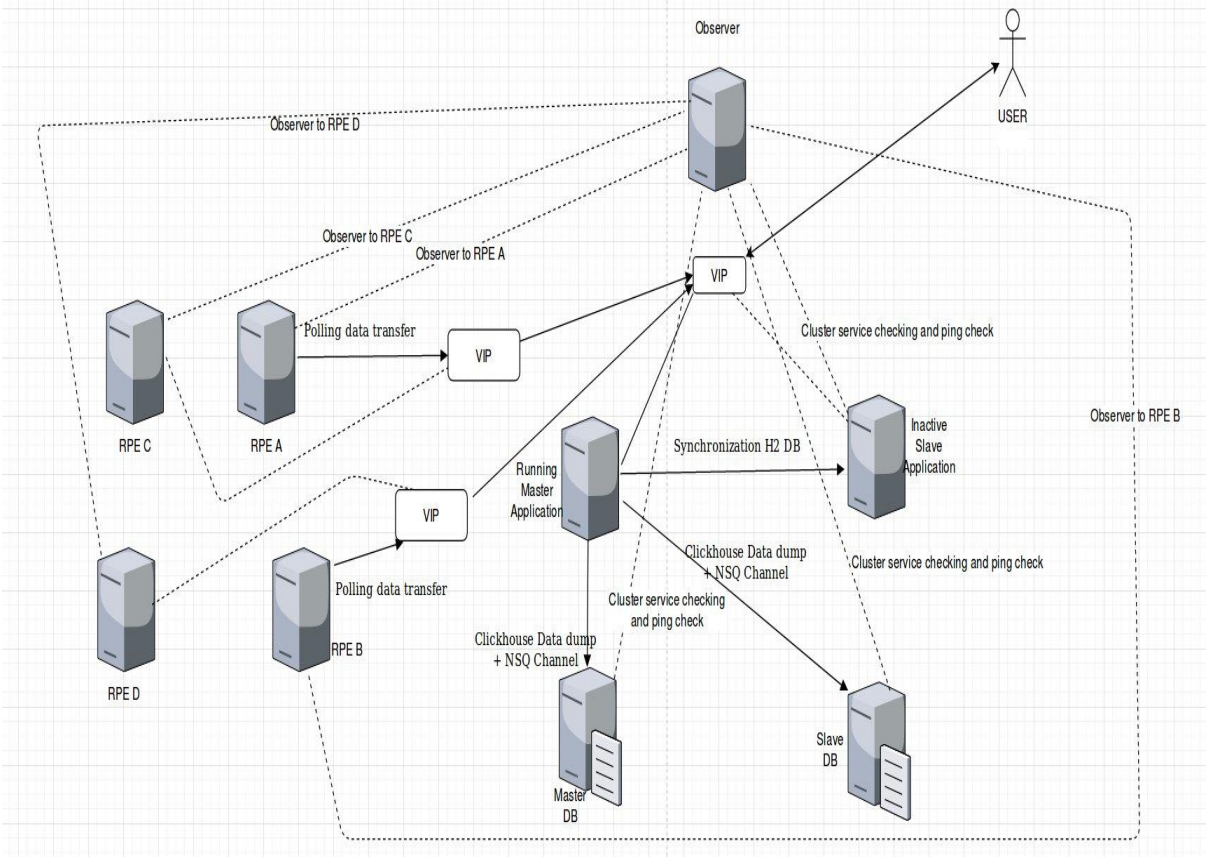
5. RPE-HA Configuration

5.1 IP Addresses

- Master Server – 172.16.8.137
- Slave Server – 172.16.8.122
- Master Data Node – 172.16.10.136
- Slave Data Node – 172.16.8.147
- Observer Utility – 172.16.10.141

- Virtual IP – 172.16.8.253
- Active RPE1 – 172.16.8.152
- Passive RPE1 – 172.16.8.211
- VIP for RPE1 – 172.16.10.251
- Active RPE2 – 172.16.8.245
- Passive RPE2 – 172.16.8.218
- VIP for RPE2 – 172.16.10.252

5.2 Diagram



5.3 Active/Passive RPE Configuration

Please do required configuration in below mention configuration file:

- rpe-conf.yml

Active RPE Configuration

```
#https rpe master host(vip of master)
host: 172.16.8.253

#https rpe master server port
port: 443

#rpe client vip host
rpe-host: 172.16.10.251

#rpe-physical-address(physical ip-address)
rpe-physical-address: 172.16.8.152

#reconnection time in seconds used to try for connection after disconnection from master server
reconnect-time: 30

#disconnection duration in hours, Must be between 1-24,default is 1
disconnect-duration: 1

#certificate file name, used to communicate between rpe master and client
certificate-file-name: keystore.jks

#used to consider rpe-client goes offline. value should be 1-60(minutes), default is 10.
rpe-status-duration: 10
~
~
~
~
```

1. Open /motadata/motadata/config/rpe-conf.yml file.
2. host: 172.16.8.253 (Master Application VIP)
3. rpe-host: 172.16.10.251 (VIP assigned to Active/Passive RPE)
4. rpe-physical-address: 172.16.8.152 (Active RPE server IP)

Passive RPE Configuration

```
#https rpe master host(vip of master)
host: 172.16.8.253

#https rpe master server port
port: 443

#rpe client vip host
rpe-host: 172.16.10.251

#rpe-physical-address(physical ip-address)
rpe-physical-address: 172.16.8.211

#reconnection time in seconds used to try for connection after disconnection from master server
reconnect-time: 30

#disconnection duration in hours, Must be between 1-24,default is 1
disconnect-duration: 1

#certificate file name, used to communicate between rpe master and client
certificate-file-name: keystore.jks

#used to consider rpe-client goes offline. value should be 1-60(minutes), default is 10.
rpe-status-duration: 10

~
~
~
```

1. Open /motadata/motadata/config/rpe-conf.yml file.
2. host: 172.16.8.253 (Master Application VIP)
3. rpe-host: 172.16.10.251 (VIP assigned to Active/Passive RPE)
4. rpe-physical-address: 172.16.8.211 (Active RPE server IP)

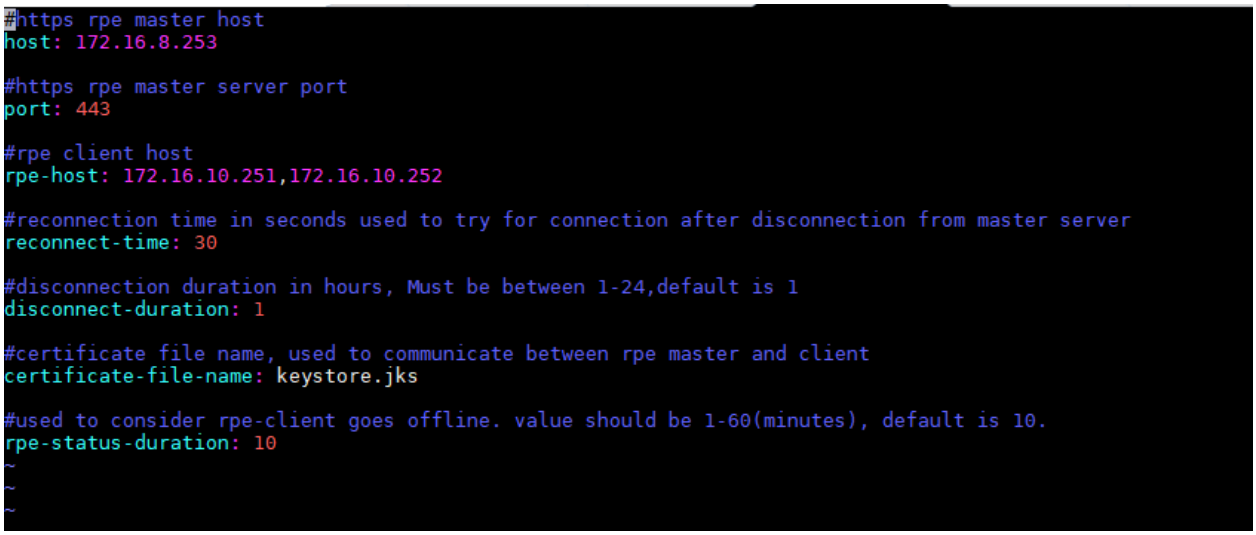
Note: Do the same configuration for multiple RPE's pair. Need to assign vip to active RPE's manually for the first time, before observer is started.

5.4 Master-Slave Server Configuration

Please do required configuration in below mention configuration file:

- motadata-conf.yml
- rpe-conf.yml

1. Open /motadata/motadata/config/motadata-conf.yml file.
2. Installation-type : 1



1. Open /motadata/motadata/config/rpe-conf.yml file.
2. host: 172.16.8.253 (Master Application VIP)
3. rpe-host: 172.16.10.251, 172.16.10.252 (VIP of RPE)

Note: Do all the above same configuration in the slave server also.

5.5 Observer server configuration

Please do required configuration in below mention configuration file:

- observer-conf.yml
- rpe-conf.yml

Observer-conf.yml

```
#schedule duration in seconds [check master host connectivity on every provided seconds] (not allowed less then 10 second)
schedule-duration: 10
#all config database hosts
config-db-host: 172.16.8.137,172.16.8.122
#all report database hosts
report-db-host: 172.16.10.136,172.16.8.147
#current system host <current-master-host>:<ui-port> | ex :[192.168.1.1:8080]
current-observer-host: 172.16.10.141
#reconnect time in seconds [consider master as down after this much seconds connectivity not available] (not allowed less then 10 second)
reconnect-time: 60
#currently running or alive as master host
current-master-host: 172.16.8.137:8080
#ssh connection port for all hosts
ssh-port: 22
#master hosts which are not alive as master <current-master-host>:<ui-port> | ex :[192.168.1.1:8080]
target-host: 172.16.8.122:8080
#rpe hosts
rpe-target-host:
#currently down hosts
down-target-host:
#delete logs after days
log-retention-days: 2
# define all ssh username
ssh-username:
# define observer ssh username
observer-ssh-username:
#define vip
vip-ip: 172.16.8.253
# sample vip-command : ifconfig ens160:2 172.16.8.251 netmask 255.255.255.0 up
vip-command: ifconfig ens160:2 172.16.8.253 netmask 255.255.255.0 up
#rpe-observer -> enable/disable to run observer for rpe(default disable)
rpe-observer-enable: 'enable'
```

1. Open /motadata/motadata/config/observer-conf.yml file.
2. rpe-observer-enable: enable (Add this parameter for observing the RPE for failover)

rpe-conf.yml

```
rpe-host-vips: 172.16.10.251,172.16.10.252
rpe-target-hosts: 172.16.8.211-ens160,172.16.10.218-ens160
max-rpe-ping-check-timeout: '3'
rpe-current-hosts: 172.16.8.152-ens160,172.16.8.245-ens160
rpe-schedule-duration: '10'
reconnect-time: '30'
max-rpe-ping-check-retry-count: '3'
current-master-host: 172.16.8.253
certificate-file-name: keystore.jks
down-target-host: ''
```

1. Open /motadata/motadata/config/rpe-conf.yml file.
2. rpe-host-vips: 172.16.10.251,172.16.10.252 (VIP of pair of Active/Passive RPE)
3. rpe-target-hosts: 172.16.8.211-ens160,172.16.10.218-ens160 (This field is for passive RPE)
4. max-rpe-ping-check-timeout: 3 (Default value is 3. Value less than 3 will not be counted)
5. rpe-current-hosts: 172.16.8.152-ens160,172.16.8.245-ens160 (This field is for active RPE)
6. rpe-schedule-duration: 10 (the heartbeat check time, at this interval the observer will go for verifying services on rpe server)
7. reconnect-time: 30 (time for which the observer will wait , before switching over)
8. max-rpe-ping-check-retry-count: 3 (Default value is 3. Value less than 3 will not be counted)
9. current-master-host: 172.16.8.253 (Master Application VIP)

Note: We need to maintain sequence in rpe-conf.yml in observer server.

6. Troubleshooting Steps

You can check the following configurations for root cause analysis of the issue. Here we are assuming that Motadata is newly deployed with HA functionality and Motadata is working fine.

Both master and slave servers are up and running.

- Motadata service should be running in **Master** machine.
- Motadata-cluster service should be running in **slave, DB1 and DB2** machine.
- Clickhouse service should be running in **DB1 and DB2** machine.

Master server goes down.

- Motadata service should be running in **Slave** machine.
- Motadata-cluster service should be running in **DB1 and DB2** machine.
- Clickhouse service should be running in **DB1 and DB2** machine.

Motadata server comes up.

- Motadata service should be running in **Slave** machine.
- Motadata-cluster service should be running in **Master, DB1 and DB2** machine.
- Clickhouse service should be running in **DB1 and DB2** machine.

Slave goes down.

- Motadata service should be running in **Master** machine.
- Motadata-cluster service should be running in **DB1 and DB2** machine.
- Clickhouse service should be running in **DB1 and DB2** machine.

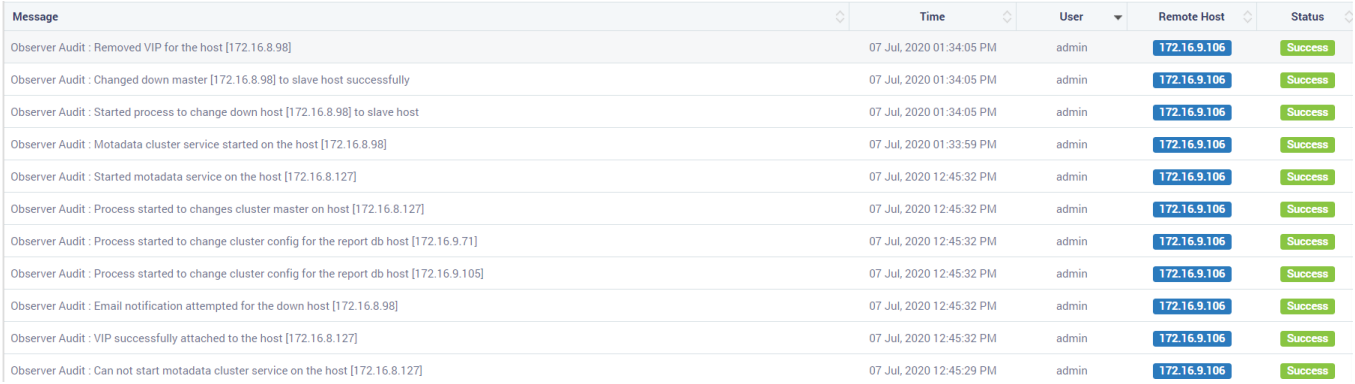
Slave comes up.

- Motadata service should be running in **Master** machine.
- Motadata-cluster service should be running in **Slave, DB1 and DB2** machine.
- Clickhouse service should be running in **DB1 and DB2** machine.

6.1 Audit Logs

- In case of any HA Fail Over or Fail Back, activity can be viewed in the Audit logs

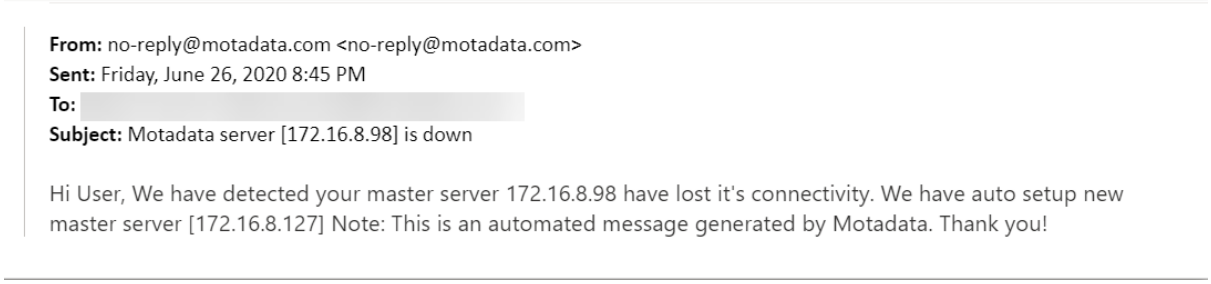
- Go to **Admin** >> **Security** >> **Audit**



Message	Time	User	Remote Host	Status
Observer Audit : Removed VIP for the host [172.16.8.98]	07 Jul, 2020 01:34:05 PM	admin	172.16.9.106	Success
Observer Audit : Changed down master [172.16.8.98] to slave host successfully	07 Jul, 2020 01:34:05 PM	admin	172.16.9.106	Success
Observer Audit : Started process to change down host [172.16.8.98] to slave host	07 Jul, 2020 01:34:05 PM	admin	172.16.9.106	Success
Observer Audit : Motadata cluster service started on the host [172.16.8.98]	07 Jul, 2020 01:33:59 PM	admin	172.16.9.106	Success
Observer Audit : Started motadata service on the host [172.16.8.127]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : Process started to changes cluster master on host [172.16.8.127]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : Process started to change cluster config for the report db host [172.16.9.71]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : Process started to change cluster config for the report db host [172.16.9.105]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : Email notification attempted for the down host [172.16.8.98]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : VIP successfully attached to the host [172.16.8.127]	07 Jul, 2020 12:45:32 PM	admin	172.16.9.106	Success
Observer Audit : Can not start motadata cluster service on the host [172.16.8.127]	07 Jul, 2020 12:45:29 PM	admin	172.16.9.106	Success

6.2 Notification

- A notification for any Failover or Backup will also be received via email to the configured Admin User



From: no-reply@motadata.com <no-reply@motadata.com>
Sent: Friday, June 26, 2020 8:45 PM
To:
Subject: Motadata server [172.16.8.98] is down

Hi User, We have detected your master server 172.16.8.98 have lost it's connectivity. We have auto setup new master server [172.16.8.127] Note: This is an automated message generated by Motadata. Thank you!

7. HA with vMotion

7.1 Pre-requisite

- VMotion feature should be configured in vmWare infrastructure.

This guide will help you to configure HA feature for single box as well as distributed architecture with the help of vMotion feature. Here,

- You don't require to configure observer, as failover and failback functionality will be taken care by vMotion.
- And On all servers, you will need to enable auto-service using below command:

  systemctl enable motadata.service

Note:

1. *For single box, use above command on Master Server and for distributed architecture, use this command for both Master as well as all RPE.*
2. We don't need to follow previous HA feature when we use above vMotion feature

Keep in touch

www.motadata.com, sales@motadata.com,

India: +91 79-2680-0900, USA: +1 408-418-5229

About Motadata

Mindarray Systems Pvt. Ltd. a global IT product company, offers state of the art affordable yet powerful product suite - Motadata consisting of Network Management & Monitoring, Log & Flow Management, and IT Service Management Platforms. The platform empowers both IT administrators and CXOs to analyze, track & resolve IT operational issues by effectively monitoring various systems and devices from multiple vendors through a unified and centralized dashboard.

Motadata is industry's first IT ops solution that truly correlates the metric, flow and log events and turns them into actionable insights. Our global customers from Telecom, Government and Enterprise domain, rely on Motadata for proactively monitor their network infrastructure.

For more information, visit www.motadata.com.

motadata