IT490 - Kaplunk

Contents

[First step 2](#_Toc8844706)

[Download and install necessary software: 2](#_Toc8844707)

[After Ubuntu installation 2](#_Toc8844708)

[Make directory and download github repository: 2](#_Toc8844709)

[Time for package management: 2](#_Toc8844710)

[RabbitMQ 2](#_Toc8844711)

[Web server 3](#_Toc8844712)

[Database 3](#_Toc8844713)

[Install php for mysql next: 3](#_Toc8844714)

[Firewall 3](#_Toc8844715)

[Setting Up Default Policies 3](#_Toc8844716)

[Allowing SSH 3](#_Toc8844717)

[Enabling UFW 3](#_Toc8844718)

[Allowing Other Connections 4](#_Toc8844719)

[Replication using Master – Slave configuration 4](#_Toc8844720)

[Login to Master Server 4](#_Toc8844721)

[Login to Slave Server 4](#_Toc8844722)

[Failover with KeepAlived 5](#_Toc8844723)

[Install Required Packages on all VMs 5](#_Toc8844724)

[Setup Keepalived on VM1 5](#_Toc8844725)

[Setup KeepAlived on VM2 6](#_Toc8844726)

[Start KeepAlived Service 6](#_Toc8844727)

[Check Virtual IPs 6](#_Toc8844728)

[DMZ 7](#_Toc8844729)

[Deployment 7](#_Toc8844730)

# First step

## Download and install necessary software:

Install Oracle VM VirtualBox 6.0

Install Ubuntu 16.04 LTS

## After Ubuntu installation

Open terminal window and type following commands:

sudo apt-get install git

sudo apt-get install gitk

sudo apt-get install php

sudo apt-get install vim

sudo apt-get install aptitude

## Make directory and download github repository:

mkdir git

cd git

git clone <https://github.com/edortu1/IT490-C.E.N.P/tree/master>

## Time for package management:

sudo aptitude

When it loads press "/"

type "php amqp" and press enter

press "n" until "php amqp" is highlighted (it might be highlighted already)

press "+"

press "g" and then "g" again

when prompted, press "q" and then hit enter to quit aptitude

Sudo apt-get install rabbitmq-server

Make startup servers

You can create a small server using PHP by following this guide: <https://medium.com/@benmorel/creating-a-linux-service-with-systemd-611b5c8b91d6>

# RabbitMQ

Sudo apt-get install sendmail

Sudo rabbitmq-plugins enable rabbitmq\_management

Open firefox, go to localhost:15672

Login with username and password guest

Create a new vhost “testHost”

Add new user “admin” and make password

Give permissions for root “/” and “testHost”

Create another user “test” with pass “test”

Create a new exchange testExchange” for “testHost” settings

Create a new queueu “testQueue” for “testHost” routing key

Click on testQueue, bind with exchange “testExchange”

Sudo apt-get install mysql-server

Sudo apt-get install python3

Sudo apt-get install python3-pip

Sudo pip install pika

Sudo pip install mysql-connector

Install a firewall with UFW between back end and front end

# Web server

Install Apache

Sudo apt update

Sudo apt install apache2

Sudo ufw app list list

Cd /var/www/html

# Database

Sudo apt-get install mysql-server

Create root password

Login with the command “sudo mysql –u root p”

Type in commands:

CREATE USER ‘admin’@’localhost’ IDENTIFIED BY ‘pass’;

GRANT ALL PRIVILEGES on \*.\* TO ‘admin’@’localhost’ WITH GRANT OPTION;

FLUSH PRIVILEGES;

## Install php for mysql next:

Sudo apt-get install php-mysql

# Firewall

## Setting Up Default Policies

sudo ufw default deny incoming

sudo ufw default allow outgoing

## Allowing SSH

sudo ufw allow ssh

sudo ufw allow 22

## Enabling UFW

sudo ufw enable

The firewall is now active, to see the rules:

sudo ufw status verbose

## Allowing Other Connections

sudo ufw allow http

sudo ufw allow https

sudo ufw allow ftp

# Replication using Master – Slave configuration

## Login to Master Server

Edit & Modify the Configuration file of MySql Server

vim /etc/mysql/mysql.conf.d/mysqld.cnf

bind-address = 127.0.0.1 #comment this line if you want to remotely access your server

Add below lines at the end of the file:

server-id = 1

log\_bin = /var/log/mysql/mysql-bin.log

log\_bin\_index =/var/log/mysql/mysql-bin.log.index

relay\_log = /var/log/mysql/mysql-relay-bin

relay\_log\_index = /var/log/mysql/mysql-relay-bin.index

Restart Mysql Server:

service mysql restart

Login to Mysql Server:

mysql -u root -p

Create a new user for Replication & specify the Password to that user:

mysql > create user 'replica'@'%' identified by 'password';

mysql > GRANT REPLICATION SLAVE ON \*.\* TO 'replica'@'%';

mysql > FLUSH PRIVILEGES;

Execute below command to view the File & Position of Master Server:

mysql > show master status;

## Login to Slave Server

Edit & Modify the Configuration file of MySql Server:

vim /etc/mysql/mysql.conf.d/mysqld.cnf

bind-address = 127.0.0.1

Add below lines at the end of the file:

server-id = 2

log\_bin = /var/log/mysql/mysql-bin.log

log\_bin\_index =/var/log/mysql/mysql-bin.log.index

relay\_log = /var/log/mysql/mysql-relay-bin

relay\_log\_index = /var/log/mysql/mysql-relay-bin.index

Restart Mysql Server:

service mysql restart

Login to Mysql Server:

mysql -u root -p

Specify the following details as given below & make sure to replace the following settings with yours.

MASTER\_HOST : IP Address of Master server

MASTER\_USER : Replication User of Master server that we had created in previous steps.

MASTER\_PASSWORD : Replication User Password of Master server that we had created in previous steps.

MASTER\_LOG\_FILE : Your Value of Master Log File of Master server.

MASTER\_LOG\_POS : Your Value of Master Log Position of Master server.

mysql > stop slave;

mysql > CHANGE MASTER TO MASTER\_HOST = 'master-ip', MASTER\_USER = 'replica', MASTER\_PASSWORD = 'password', MASTER\_LOG\_FILE = 'mysql-bin.000001', MASTER\_LOG\_POS = 753;

mysql > start slave;

# Failover with KeepAlived

## Install Required Packages on all VMs

sudo apt-get install linux-headers-$(uname -r)

Install Keepalived:

sudo apt-get install keepalived

## Setup Keepalived on VM1

vim /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

notification\_email {

sysadmin@mydomain.com

support@mydomain.com

}

notification\_email\_from VM1@mydomain.com

smtp\_server localhost

smtp\_connect\_timeout 30

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 101

priority 101

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

YOUR IP GOES HERE

}

}

## Setup KeepAlived on VM2

vim /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

notification\_email {

sysadmin@mydomain.com

support@mydomain.com

}

notification\_email\_from lb2@mydomain.com

smtp\_server localhost

smtp\_connect\_timeout 30

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 101

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

YOUR IP GOES HERE

}

}

## Start KeepAlived Service

sudo service keepalived start

## Check Virtual IPs

ip addr show eth

# DMZ

Sudo apt-get install python3

Sudo apt-get install python3-pip

Sudo pip install pika

Sudo pip install mysql-connector

# Deployment

Follow General VM instructions

Copy deployServer.service to /lib/systemd/system/ and enable and start deployServer

Mkdir versions

Create database versiondb;

Insert tables from createTable.txt into versiondb

Insert 0 versions for fe be dmz and deploy into deployTable.

Insert ip addresses for all machines based on sample in createTable.txt

Cd /etc/apache2/mods-available/

Sudo a2enmod

Proxy\_balancer

Proxy\_http

Proxy\_lbmethod\_byrequests

Copy 000-default.conf to /etc/apache2/sites-available

Restart apache