Apache快速设置虚拟主机脚本

目录

[Apache快速设置虚拟主机脚本 1](#_Toc517777461)

[一、 项目背景意义和需求分析 1](#_Toc517777462)

[1. 背景意义： 1](#_Toc517777463)

[2. 需求分析： 2](#_Toc517777464)

[二、 术语解释 2](#_Toc517777465)

[1. 虚拟主机： 2](#_Toc517777466)

[2. Lamp环境： 2](#_Toc517777467)

[三、 系统搭建 2](#_Toc517777468)

[1. DHCP服务器搭建 3](#_Toc517777469)

[2. DNS服务器搭建 3](#_Toc517777470)

[3. APACHE MYSQL服务器搭建 5](#_Toc517777471)

[4. 控制APACHE服务器虚拟主机脚本 7](#_Toc517777472)

[四、 功能说明 14](#_Toc517777473)

[1. 显示apache网页服务器状态、虚拟主机配置和firewall防火墙状态 15](#_Toc517777474)

[2. 添加虚拟主机 15](#_Toc517777475)

[3. 移除虚拟主机 16](#_Toc517777476)

[4. 修改虚拟主机 18](#_Toc517777477)

[5. 打开防火墙并且添加http服务永久生效 21](#_Toc517777478)

[6. 打开防火墙并且移除http服务永久生效 23](#_Toc517777479)

[7. 退出脚本 23](#_Toc517777480)

[五、 总结与展望 24](#_Toc517777481)

[总结： 24](#_Toc517777482)

[展望： 24](#_Toc517777483)

[六、 参考文献 24](#_Toc517777484)

## 项目背景意义和需求分析

### 背景意义：

Apache虚拟主机就是在一个Apache服务器上配置多个虚拟主机，实现一个服务器提供多站点服务，其实就是访问同一个服务器上的不同目录。本项目搭建了dhcp服务器，分配固定ip：192.168.170.101给apache服务器，分配固定ip：192.168.170.102给dns服务器。dns服务器提供域名解析。

### 需求分析：

当修改添加一台虚拟主机到Apache服务器时，常用的方式为修改配置文件，而修改配置文件容易出错，导致整个服务器启动不了。而使用shell提前编写规则，修改配置文件，等待用户输入参数后，大大减小手动输入出错的概率。为了便于使用，脚本还提供apache服务器状态查看，防火墙配置功能。

## 术语解释

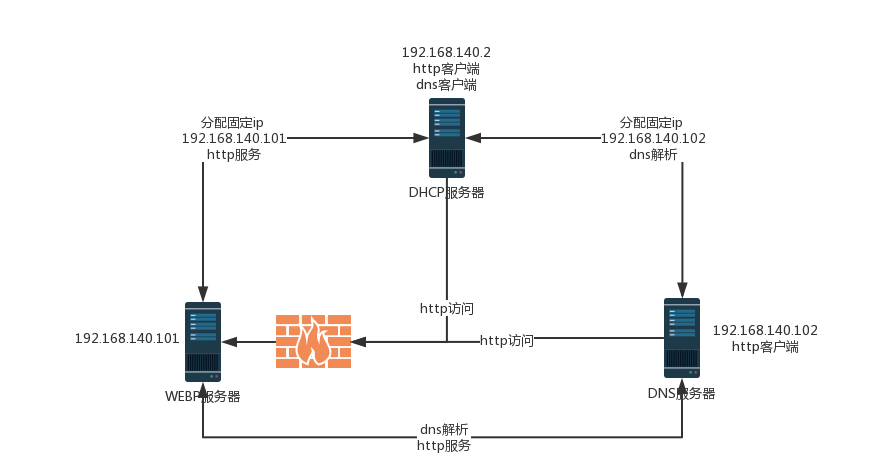
### 虚拟主机：

Apache虚拟主机就是在一个Apache服务器上配置多个虚拟主机，实现一个服务器提供多站点服务，访问同一个服务器上的不同目录。

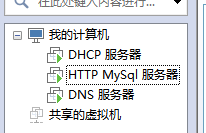
### Lamp环境：

本项目使用的是centos7+Apache+ mariadb+php

## 系统搭建



网络系统拓补图



VM虚拟机

### DHCP服务器搭建

安装

yum -y install dhcp

配置文件

vim /etc/dhcp/dhcpd.conf

ault-lease-time 600;

max-lease-time 7200;

subnet 192.168.140.0 netmask 255.255.255.0 {

range 192.168.140.10 192.168.140.20;

option routers 192.168.140.1;

option broadcast-address 192.168.255.255;

host httpserve {

hardware ethernet 00:0c:29:80:25:bf;

fixed-address 192.168.140.101;

}

host dnsserve {

hardware ethernet 00:0c:29:53:c0:1a;

fixed-address 192.168.140.102;

}

}

### DNS服务器搭建

安装

yum -y install bind

配置文件

vim /etc/named.conf

options {

directory "/var/named";

dump-file "/var/named/data/cache\_dump.db";

statistics-file "/var/named/data/named\_stats.txt";

};

zone "jzg.com" IN {

type master;

file "jzg.com";

};

zone "140.168.192.in-addr.arpa" {

type master;

file "140.168.192";

};

zone "niuini.com" IN {

type master;

file "niuini.com";

};

正向解析文件

vim /var/named/jzg.com

86400

jzg.com. IN SOA dns.jzg.com. adminemail.jzg.com (

20180511; serial

1H ; refresh

15M ; retry

1W ; expire

1D ) ; minimum

jzg.com. IN NS dns.jzg.com.

www IN A 192.168.140.101

dns IN A 192.168.140.102

bbs IN A 192.168.140.101

niuini IN A 192.168.140.101

jzg.com. IN A 192.168.140.101

ftp IN CNAME www

other IN A 192.168.140.103

正向解析文件

vim /var/named/niuini.com

$TTL 86400

niuini.com. IN SOA dns.niuini.com. adminemail.jzg.com (

20180511; serial

1H ; refresh

15M ; retry

1W ; expire

1D ) ; minimum

niuini.com. IN NS dns.niuini.com.

www IN A 192.168.140.101

dns IN A 192.168.140.102

niuini.com. IN A 192.168.140.101

ftp IN CNAME www

other IN A 192.168.140.103

反向解析文件

$TTL 86400

@ IN SOA 140.168.192.in-addr.arpa. root.fqnu.org (

20180511 ; serial

1H ; refresh

15M ; retry

1W ; expire

1D ) ; minimum

@ IN NS dns.jzg.com.

101 IN PTR www.jzg.com.

101 IN PTR jzg.com.

101 IN PTR www.niuini.com.

101 IN PTR niuini.com.

102 IN PTR dns.jzg.com.

103 IN PTR other.jzg.com.

### APACHE MYSQL服务器搭建

安装

yum -y install httpd mariadb mariadb-server php php-mysql php-gd php-xml

配置文件

# 根目录

ServerRoot "/etc/httpd"

Listen 80

# 超时时间（秒）

Timeout 400

Include conf.modules.d/\*.conf

User apache

Group apache

ServerAdmin admin@niuini.com

# 主机地址

ServerName 192.168.140.101:80

<Directory />

AllowOverride none

Require all denied

</Directory>

# Apache 文档根目录

DocumentRoot "/var/www/html"

<Directory "/var/www">

AllowOverride None

# Allow open access:

Require all granted

</Directory>

<Directory "/var/www/html">

# Indexes 允许目录浏览

# Options Indexes FollowSymLinks

Options Indexes FollowSymLinks

AllowOverride None

Require all granted

</Directory>

<IfModule dir\_module>

DirectoryIndex index.html index.php

</IfModule>

<Files ".ht\*">

Require all denied

</Files>

ErrorLog "logs/error\_log"

LogLevel warn

<IfModule log\_config\_module>

LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined

LogFormat "%h %l %u %t \"%r\" %>s %b" common

<IfModule logio\_module>

# You need to enable mod\_logio.c to use %I and %O

LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio

</IfModule>

CustomLog "logs/access\_log" combined

</IfModule>

<IfModule alias\_module>

ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

</IfModule>

<Directory "/var/www/cgi-bin">

AllowOverride None

Options None

Require all granted

</Directory>

<IfModule mime\_module>

AddType text/html .shtml

AddOutputFilter INCLUDES .shtml

</IfModule>

AddDefaultCharset UTF-8

<IfModule mime\_magic\_module>

MIMEMagicFile conf/magic

</IfModule>

IncludeOptional conf.d/\*.conf

NameVirtualHost \*:80

<VirtualHost \*:80>

DocumentRoot /var/www/html/

ServerName jzg.com

</VirtualHost>

<VirtualHost \*:80>

DocumentRoot /var/www/html2/

ServerName niuini.com

</VirtualHost>

<VirtualHost \*:80>

DocumentRoot /var/www/html2/

ServerName www.niuini.com

</VirtualHost>

### 控制APACHE服务器虚拟主机脚本

#!/bin/bash

# this program is control Apache virtual host server

# Date:2018-06-23

# Version:1.0

# Author: jiang zhi gang

# 全局变量

ConfRootDir=""

AddVirtalConfFile(){

# 匹配行 取第二个 删除引号

ConfRoot=`awk '/^ServerRoot/' /etc/httpd/conf/httpd.conf| awk '{print $2}'| sed 's/"//g'`

ConfDir=`awk '/^IncludeOptional/' /etc/httpd/conf/httpd.conf | awk '{print $2}'| awk -F"/" '{print $1}'`

Virtual="virtual.conf"

VirtalConfFile="$ConfRoot/$ConfDir/$Virtual"

echo "Virtal Host Conf File is virtual.conf Located In $ConfRoot/$ConfDir"

PrintLine

# 查询是否存在虚拟机配置文件

if [ ! -f "$VirtalConfFile" ]

then

echo "No $Virtual File, Will Creat It!"

touch "$VirtalConfFile"

fi

}

PrintLine(){

echo "--------------------------------------------------------------------"

}

# 显示配置文件中的虚拟主机

ExistingVirtualHost(){

printf "The virtual Host That Exists In The Conf file: \n\n"

# \s空格 任意空格开头+ServerName 的行 ServerName结果集

Existing=`awk '/^\s\*ServerName/' $VirtalConfFile `

if [ -z "$Existing" ]

then

echo "Conf File Is Empty!"

PrintLine

return 0

else

# 输出匹配的当前行和下一行

awk '/^\s\*ServerName/{printf ""$0"\n";getline;printf ""$0"\n\n";}' $VirtalConfFile

PrintLine

return 1

fi

}

ApacheStatus(){

PrintLine

echo "Show Virtual Host And Apache Status , Firewall Status:"

AddVirtalConfFile

ExistingVirtualHost

#显示apache相关配置信息

echo "Apache Status:"

systemctl status httpd |grep Active

# Net Stat

REQUEST\_NUM=`systemctl status httpd | grep Status | awk '{print $4}'| sed 's/;//'`

echo "Request Number: $REQUEST\_NUM"

NETSTAT=`netstat -tunpl | grep httpd`

printf "Net Stat: \n $NETSTAT\n"

PrintLine

# 显示Firewall相关配置信息

echo "Firewall Status:"

systemctl status firewalld |grep Active

DefaultZone=`firewall-cmd --get-default-zone`

FirewallServices=`firewall-cmd --zone=$ActiveZones --list-all | grep services |awk -F":" '{print $2}'`

echo "Default Zone:$DefaultZone"

echo "Services:$FirewallServices"

}

AddVirtual(){

PrintLine

echo "Add Virtual Host:"

AddVirtalConfFile

NameVirtualHost=`awk '/^NameVirtualHost/' /etc/httpd/conf/httpd.conf| awk '{print $2}'`

# 允许本机所有80端口网络通过 基于域名的虚拟主机

if [ "$NameVirtualHost" != "\*:80" ]

then

echo "Add NameVirtualHost \*:80 >> /etc/httpd/conf/httpd.conf"

echo "NameVirtualHost \*:80" >> /etc/httpd/conf/httpd.conf

fi

ExistingVirtualHost $VirtalConfFile

read -p "Please input ServerName( www.example.com ):" ServerName

# 输入ServerName是否为空

if [ -z $ServerName ]

then

echo "Please Input ServerName"

else

read -p "Please input DocumentRoot( /var/www/html/ ):" DocumentRoot

# 输入DocumentRoot是否为空

if [ -z $DocumentRoot ]

then

echo "Please Input DocumentRoot"

else

# 写配置文件

(

cat <<EOF

# ShellCript Creat ServerName $ServerName Begin

<VirtualHost \*:80>

ServerName $ServerName

DocumentRoot $DocumentRoot

</VirtualHost>

# ShellCript Creat ServerName $ServerName End

EOF

) >> $VirtalConfFile

# 写配置文件

PrintLine

echo "Add Succeed!"

cat $VirtalConfFile

systemctl restart httpd

fi

fi

}

RemoveVirtual(){

PrintLine

echo "Remove Virtual Host:"

AddVirtalConfFile

ExistingVirtualHost $VirtalConfFile

# 配置文件是否为空

if [ $? == "0" ]

then

echo "No Virtual Host Can Be Removed!"

else

read -p "Please Select ServerName Remove It:" ServerName

result=`cat $VirtalConfFile | grep "# ShellCript Creat ServerName $ServerName Begin"`

# 输入是否为空

if [ -z $ServerName ]

then

echo "Please Input ServerName"

# 输入查询结果是否为空

elif [ -z "$result" ]

then

echo "Please Input Existing ServerName"

# 输入正确

else

sed -i "/# ShellCript Creat ServerName $ServerName Begin/,/# ShellCript Creat ServerName $ServerName End/d" $VirtalConfFile

PrintLine

echo "Remove Succeed!"

cat $VirtalConfFile

systemctl restart httpd

fi

fi

}

ModifyVirtual(){

PrintLine

echo "Modify Virtual Host:"

AddVirtalConfFile

ExistingVirtualHost $VirtalConfFile

# 配置文件是否为空

if [ $? == "0" ]

then

echo "No Virtual Host Can Be Modify !"

else

read -p "Please Select ServerName Modify It:" ServerName

result=`cat $VirtalConfFile | grep "# ShellCript Creat ServerName $ServerName Begin"`

# 输入是否为空

if [ -z "$ServerName" ]

then

echo "Please Input ServerName"

# 输入查询结果是否为空

elif [ -z "$result" ]

then

echo "Please Input Existing ServerName"

# 输入正确

else

# 显示要修改的内容

sed -n "/# ShellCript Creat ServerName $ServerName Begin/,/# ShellCript Creat ServerName $ServerName End/p" $VirtalConfFile

PrintLine

printf "What to Modify ? \n"

MENU "1" "ServerName:"

MENU "2" "DocumentRoot:"

MENU "3" "Back:"

read -p "please select a function(1-3):" U\_SELECT

case $U\_SELECT in

1)

read -p "Please Input New ServerName:" NewServerName

if [ -z "$NewServerName" ]

then

echo "Please Input NewServerName!"

else

sed -i "s/ServerName $ServerName/ServerName $NewServerName/g" $VirtalConfFile

PrintLine

echo "Modify ServerName Succeed!"

cat $VirtalConfFile

systemctl restart httpd

fi

;;

2)

read -p "Please Input New DocumentRoot:" NewDocumentRoot

if [ -z "$NewDocumentRoot" ]

then

echo "Please Input NewDocumentRoot!"

else

# sed 中有/ 需要添加转义字符\

NewDocumentRoot=$(echo $NewDocumentRoot | sed 's/\//\\\//g')

# echo $NewDocumentRoot

sed -i "/^\s\*ServerName $ServerName/{n;s/DocumentRoot.\*/DocumentRoot $NewDocumentRoot/;}" $VirtalConfFile

PrintLine

echo "Modify DocumentRoot Succeed!"

cat $VirtalConfFile

systemctl restart httpd

fi

;;

3)

echo "Back..."

;;

\*)

echo "Please Select 1-3"

;;

esac

fi

fi

}

# 关闭Selinux 打开Firewalld

CloseSeOpenFw(){

SeStatus=`getenforce`

echo "Selinux Status:$SeStatus"

if [ $SeStatus != "Disabled" ]

then

read -p "Do You Want To Close Selinux? Y/N(suggest closing it): " Close

if [ $Close == "y" && $Close == "Y"]

then

sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

sed -i 's/SELINUX=permissive/SELINUX=disabled/' /etc/selinux/config

# 临时关闭SELinux

setenforce 0

echo "Close Selinux Succeed!"

else

echo "Selinux is $SeStatus Do't Close It!"

fi

fi

FWStatus=`firewall-cmd --stat`

echo "Firewalld Status:$FWStatus"

if [ $FWStatus != "running" ]

then

read -p "Do You Want To Open Firewalld? Y/N(suggest open it): " Close

if [ $Close == "y" && $Close == "Y"]

then

systemctl restart firewalld

echo "Open Firewalld Succeed!"

return 1

else

echo "Selinux is $FWStatus Do't Open It!"

return 0

fi

fi

return 1

}

Addhttp(){

CloseSeOpenFw

if [ $? == "0" ]

then

echo "Please Open Firewalld !"

else

DefaultZone=`firewall-cmd --get-default-zone`

FirewallServices=`firewall-cmd --zone=$ActiveZones --list-all | grep services |awk -F":" '{print $2}'`

echo "Default Zone:$DefaultZone"

echo "Services:$FirewallServices"

Result=`echo $FirewallServices | awk /http/ `

if [ -z "$Result" ]

then

echo "Add Http Service:"

firewall-cmd --zone="$DefaultZone" --add-service=http --permanent

firewall-cmd --zone="$DefaultZone" --add-service=http

else

echo "Http Services Already Exist In $DefaultZone Zone"

fi

fi

}

Removehttp(){

CloseSeOpenFw

if [ $? == "0" ]

then

echo "Please Open Firewalld !"

else

DefaultZone=`firewall-cmd --get-default-zone`

FirewallServices=`firewall-cmd --zone=$ActiveZones --list-all | grep services |awk -F":" '{print $2}'`

echo "DefaultZone:$DefaultZone"

echo "Services:$FirewallServices"

Result=`echo $FirewallServices | awk /http/ `

if [ -z "$Result" ]

then

echo "Http Services NO Already Exist In $DefaultZone Zone"

else

echo "Remove Http Service:"

firewall-cmd --zone="$DefaultZone" --remove-service=http --permanent

firewall-cmd --zone="$DefaultZone" --remove-service=http

fi

fi

}

HINT(){

read -p "Press Enter to continue:"

}

MENU(){

if [ $1 == "1" ]

then

printf "+----------------------------------------+\n"

fi

printf "|%2s. %-35s |\n" "$1" "$2"

printf "+----------------------------------------+\n"

}

while true

do

clear

printf " Control Apache virtual Host Server\n"

MENU "1" "Apache And Firewalld Status:"

MENU "2" "Add Virtual Host:"

MENU "3" "Remove Virtual Host:"

MENU "4" "Modify Virtual Host:"

MENU "5" "Open Firewalld --> Add http:"

MENU "6" "Open Firewalld --> Remove http:"

MENU "7" "Exit Script:"

read -p "please select a function(1-7):" U\_SELECT

case $U\_SELECT in

1)

ApacheStatus

HINT

;;

2)

AddVirtual

HINT

;;

3)

RemoveVirtual

HINT

;;

4)

ModifyVirtual

HINT

;;

5)

Addhttp

HINT

;;

6)

Removehttp

HINT

;;

7)

exit

;;

\*)

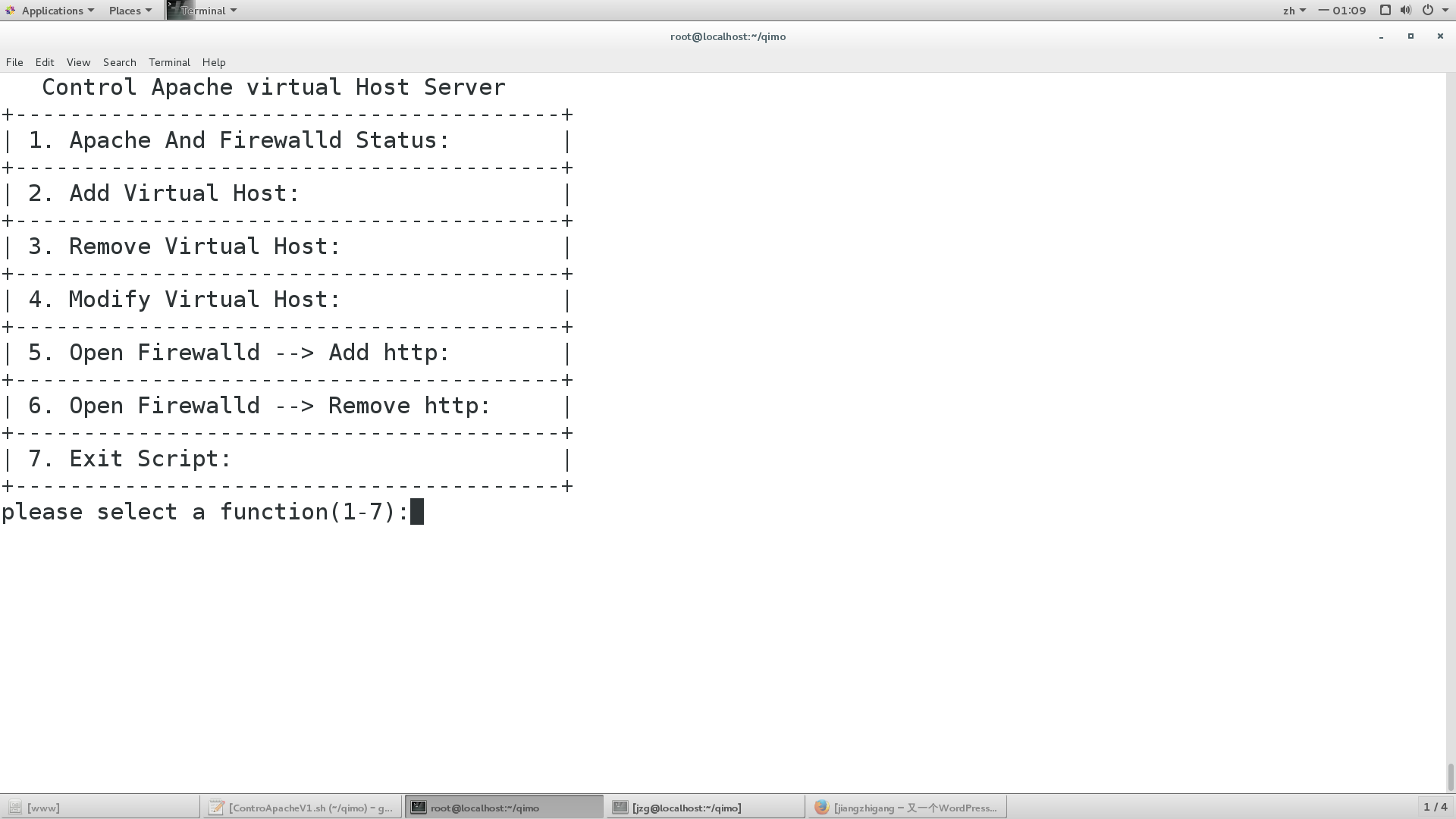
read -p "Please Select 1-7,Press Enter to continue:"

esac

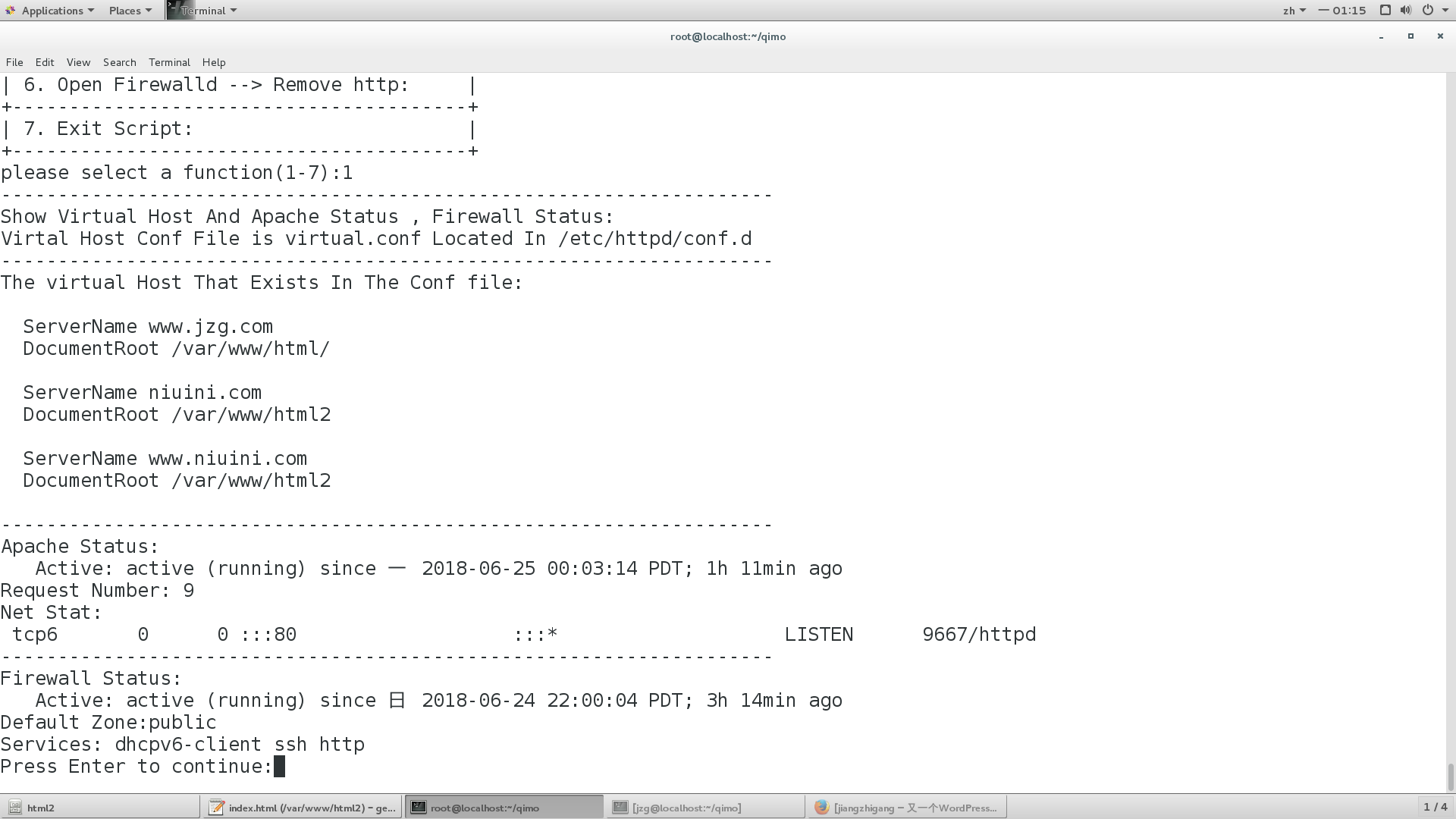
done

## 功能说明

**主菜单**



### 显示apache网页服务器状态、虚拟主机配置和firewall防火墙状态

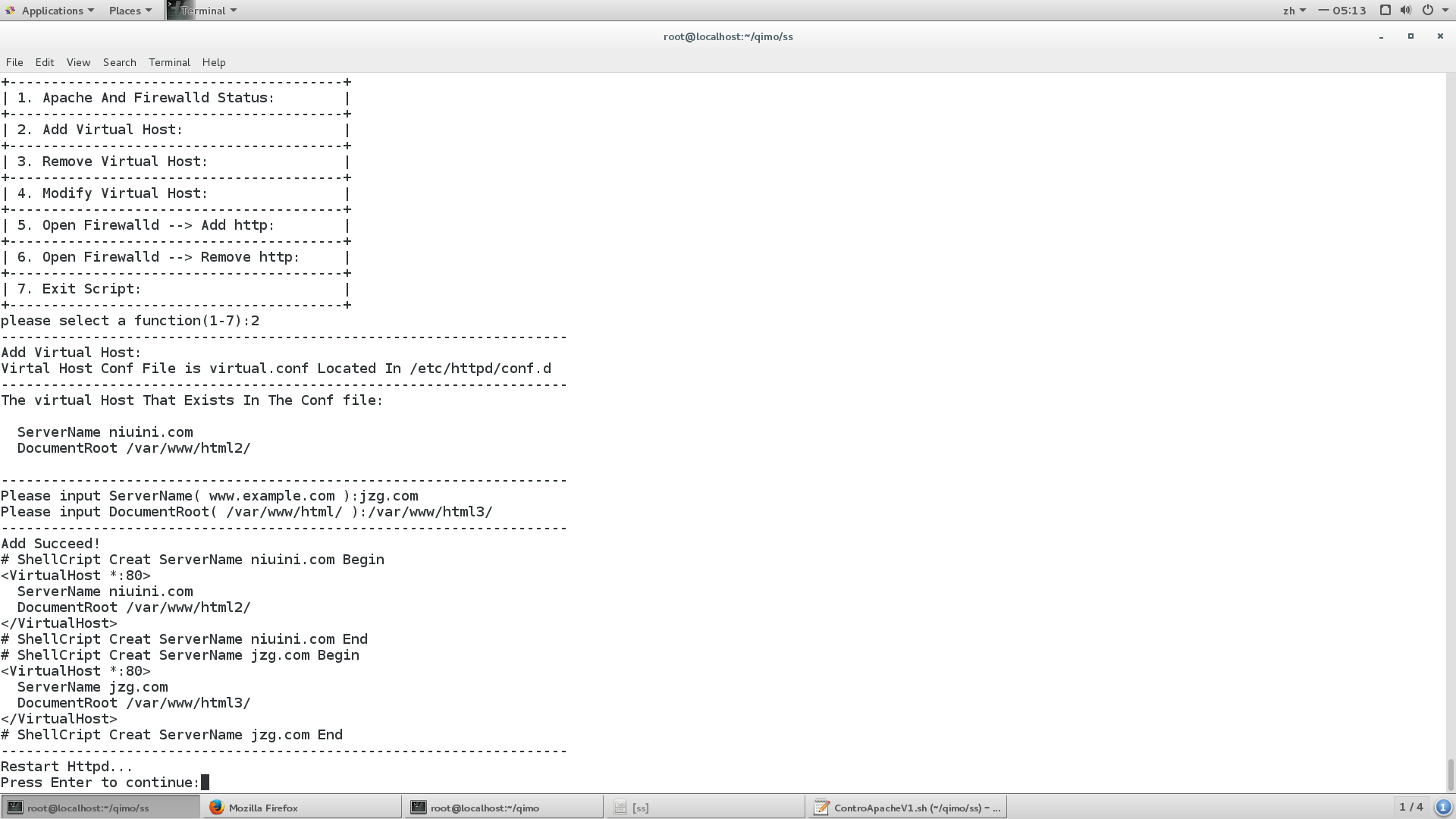
****

### 添加虚拟主机

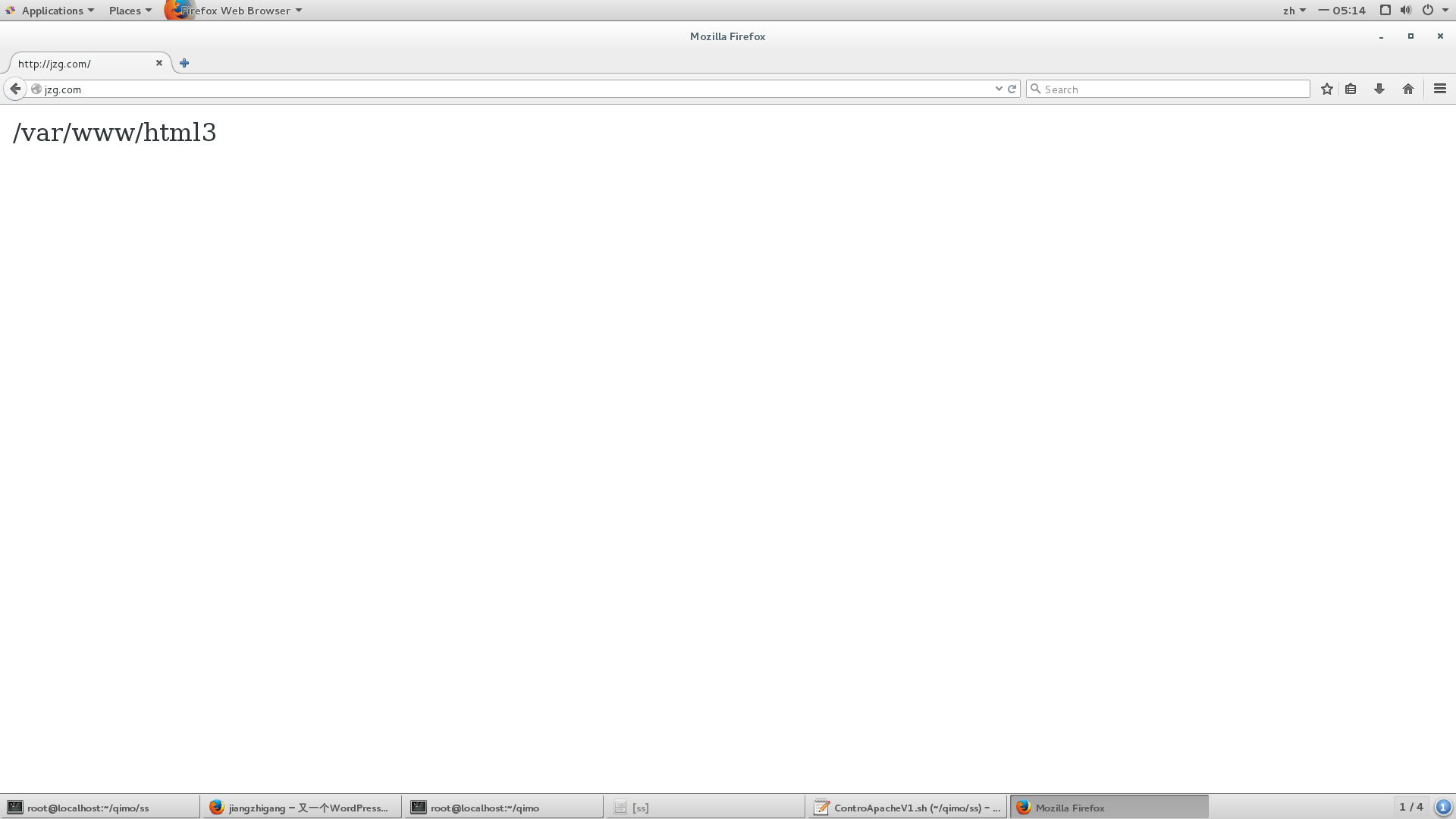
输入添加的主机域名和目录 输入jzg.com 目录 /var/www/html3/

****

添加成功 重启服务器

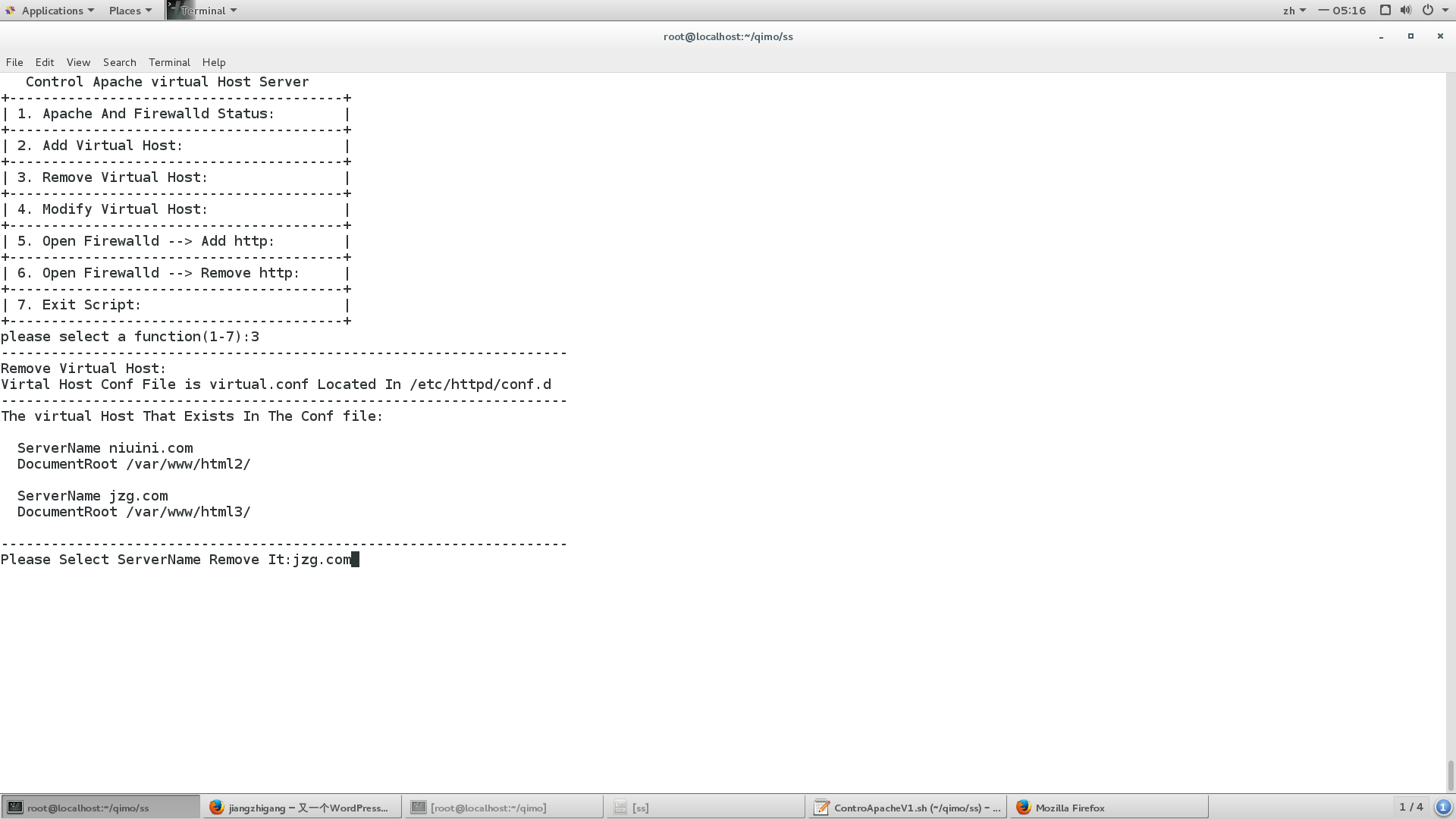
****

访问成功 /var/www/html3/index.html

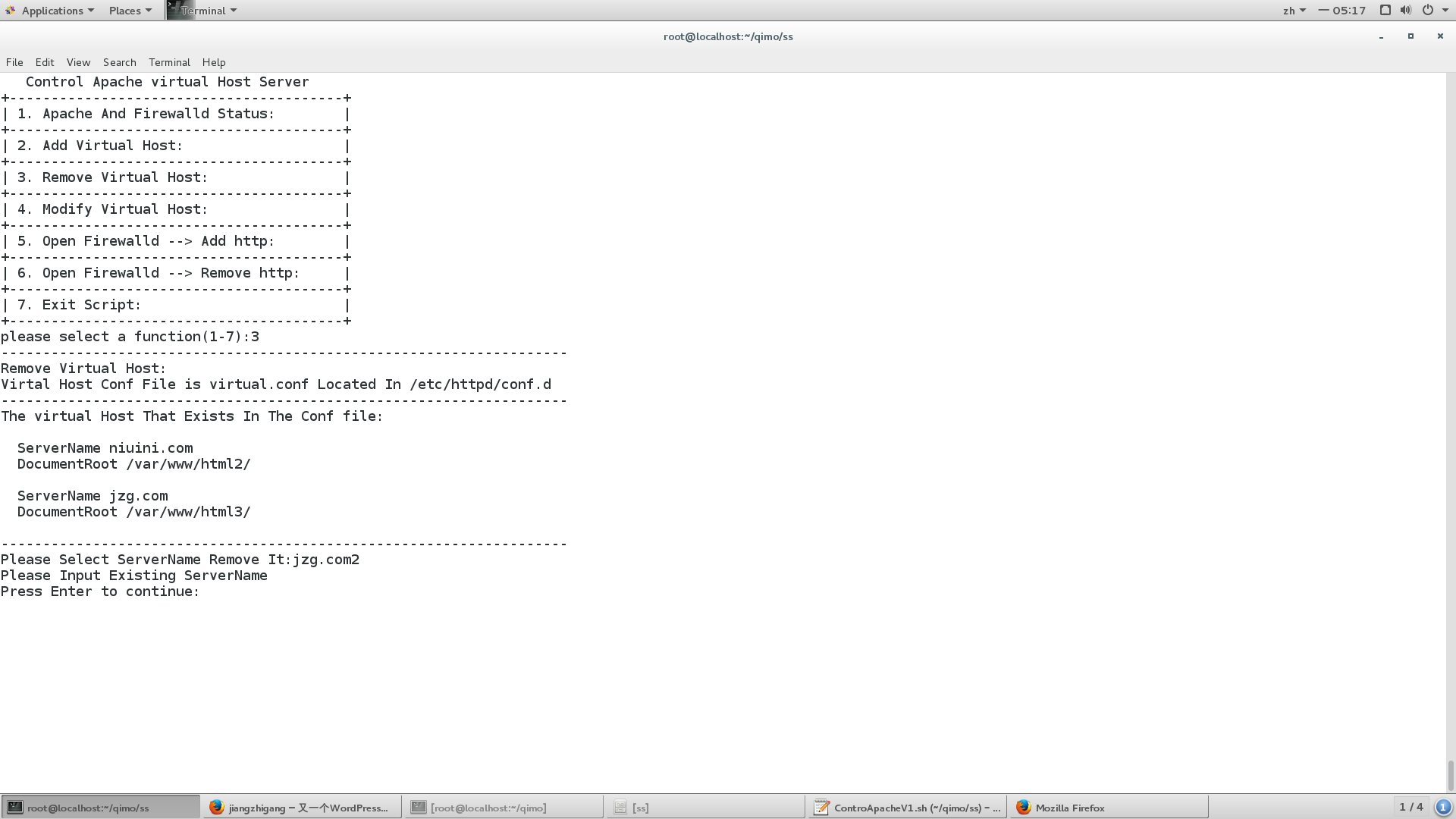
****

### 移除虚拟主机

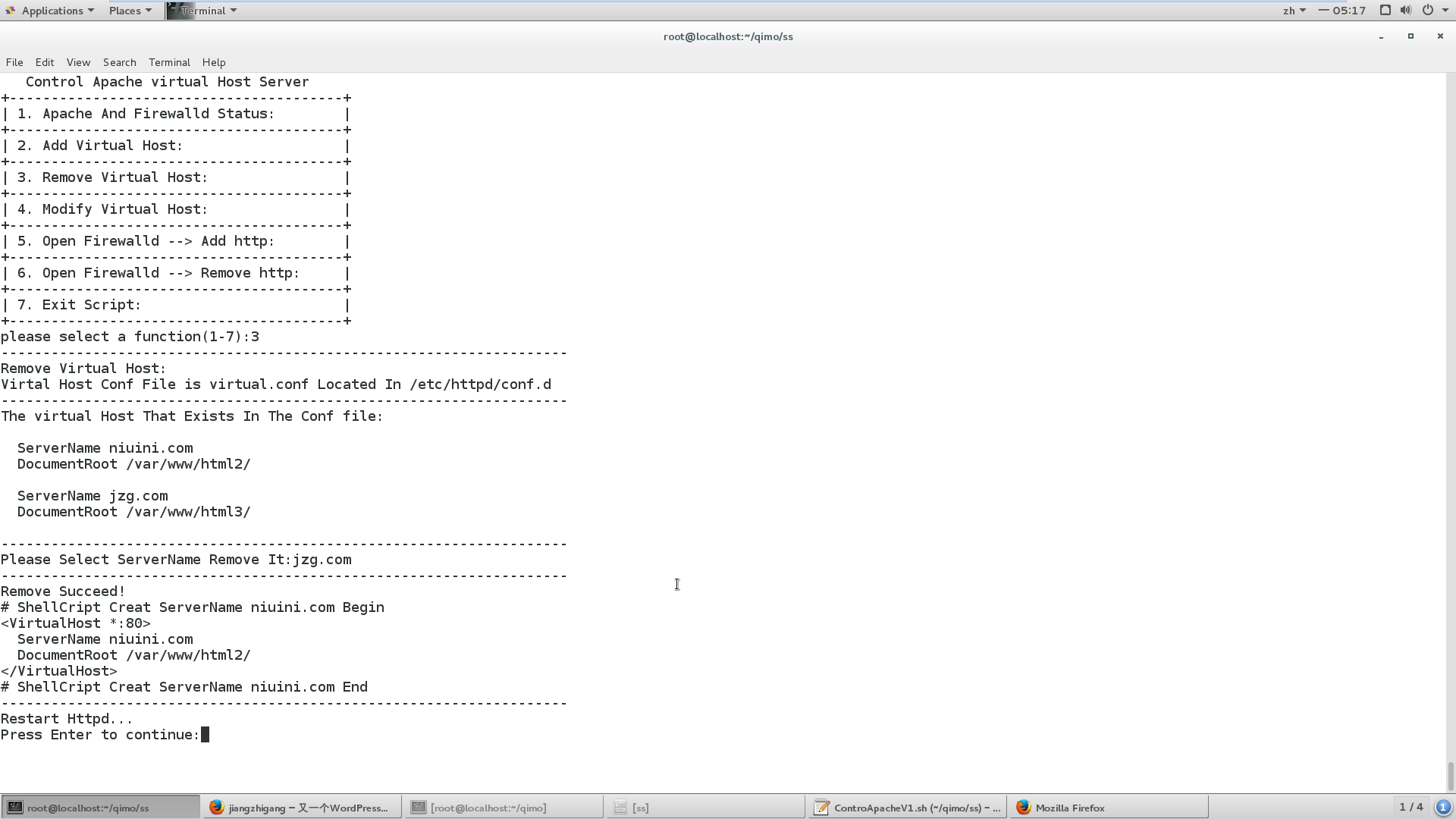
输入需要移除的主机域名

****

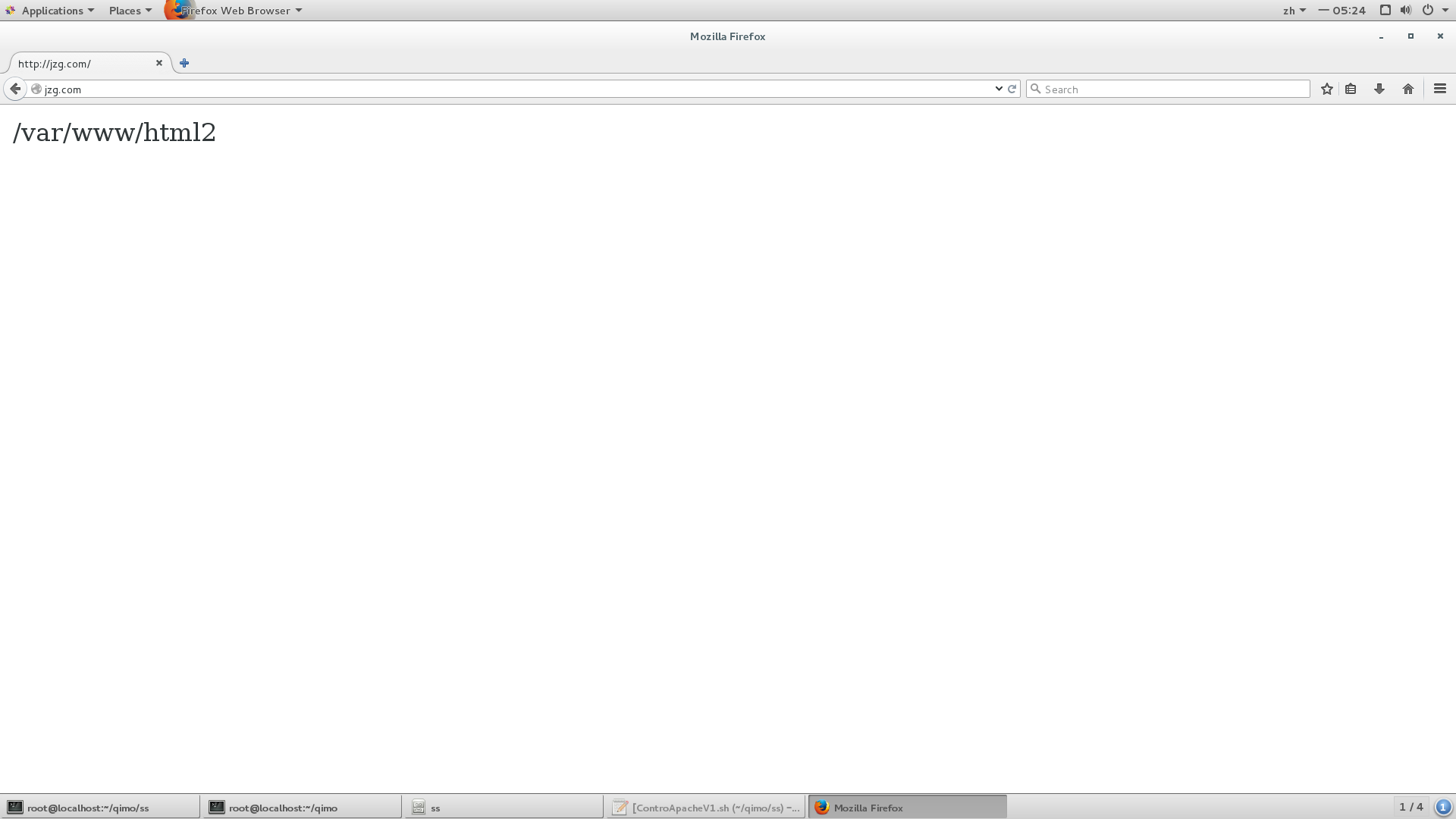
输入不存在配置文件中的域名会提示重新输入

****

输入正确，移除成功 重启服务器

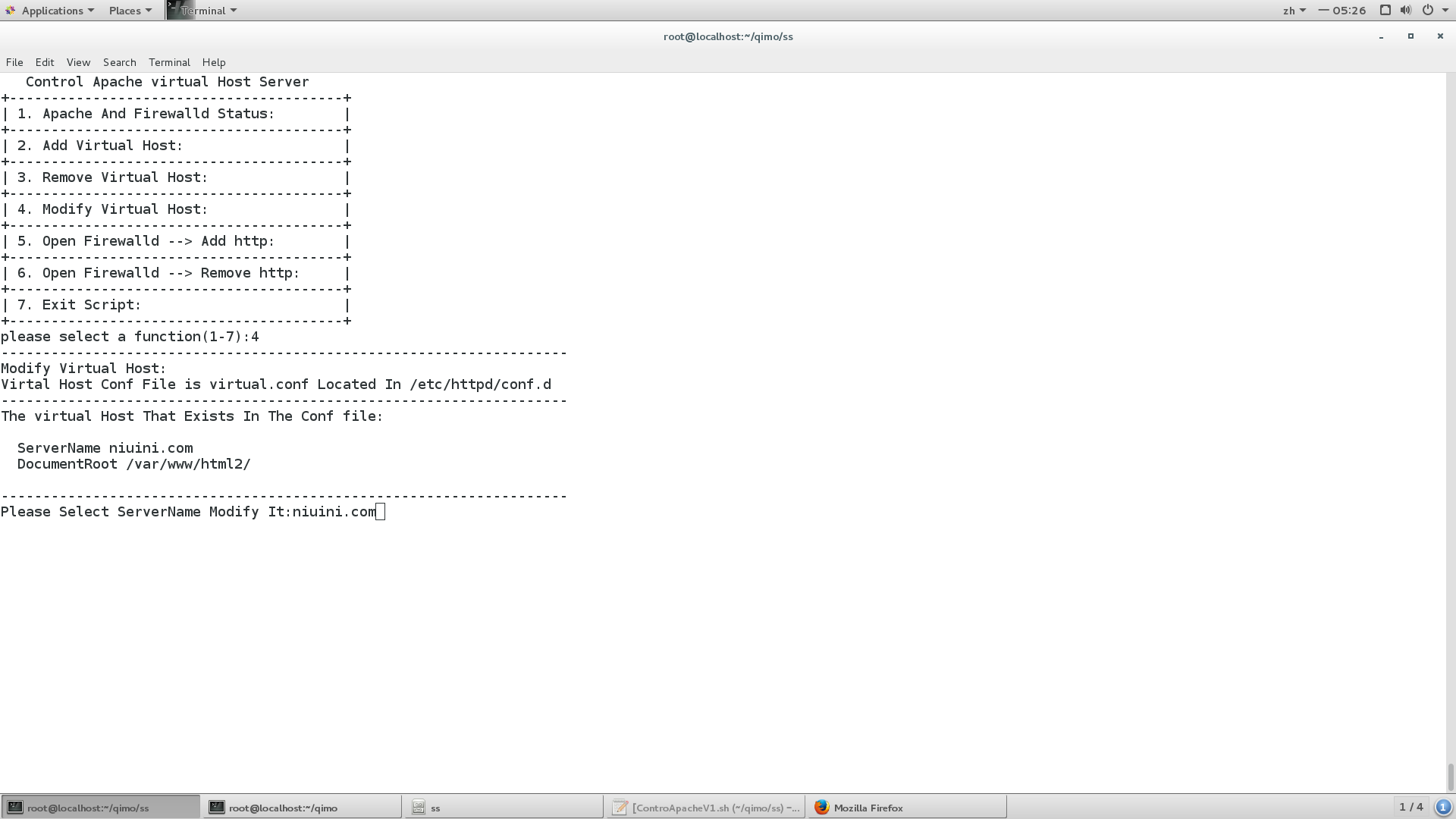
****

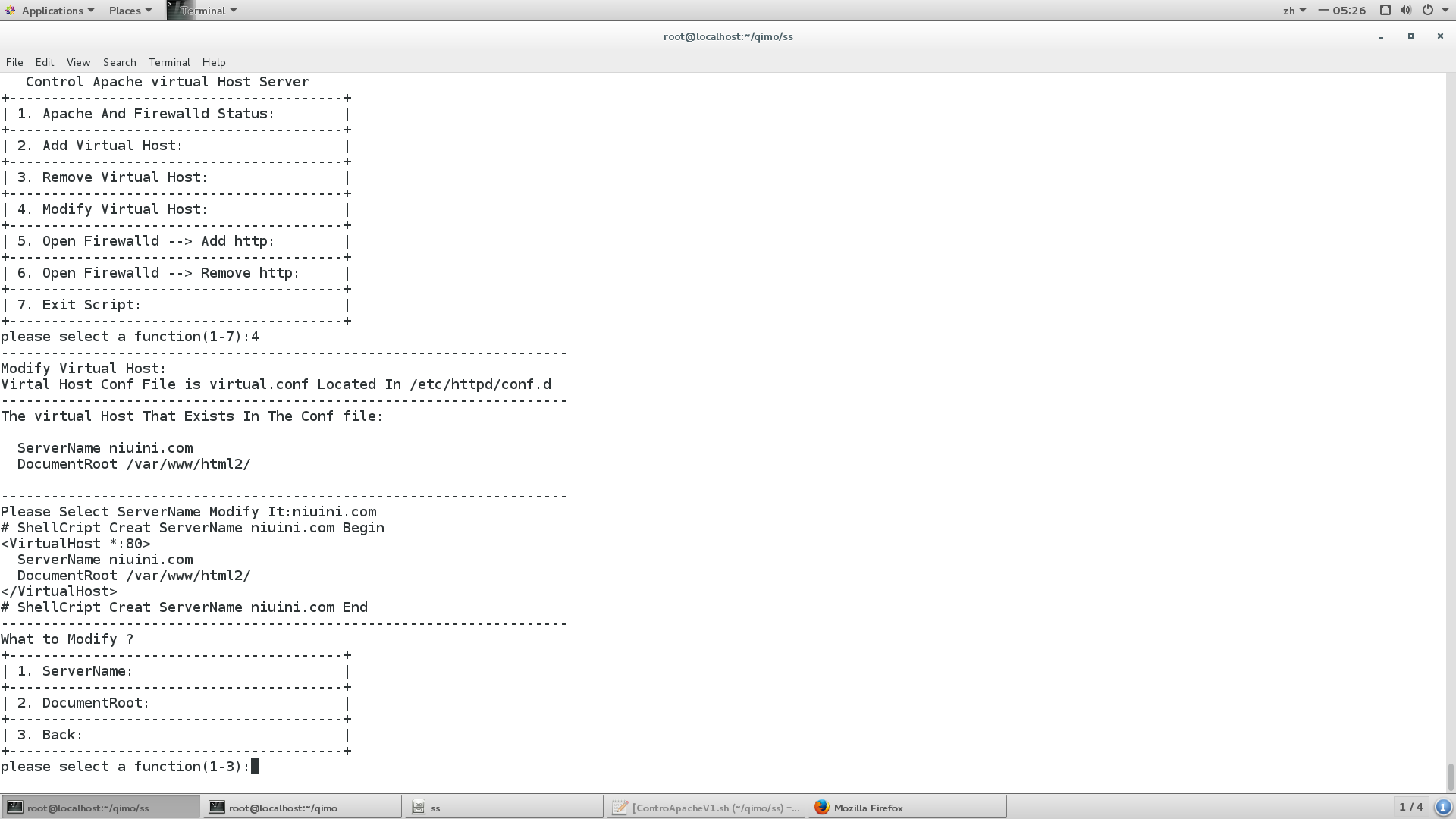
移除后 以第一个虚拟主机的目录为默认根目录

****

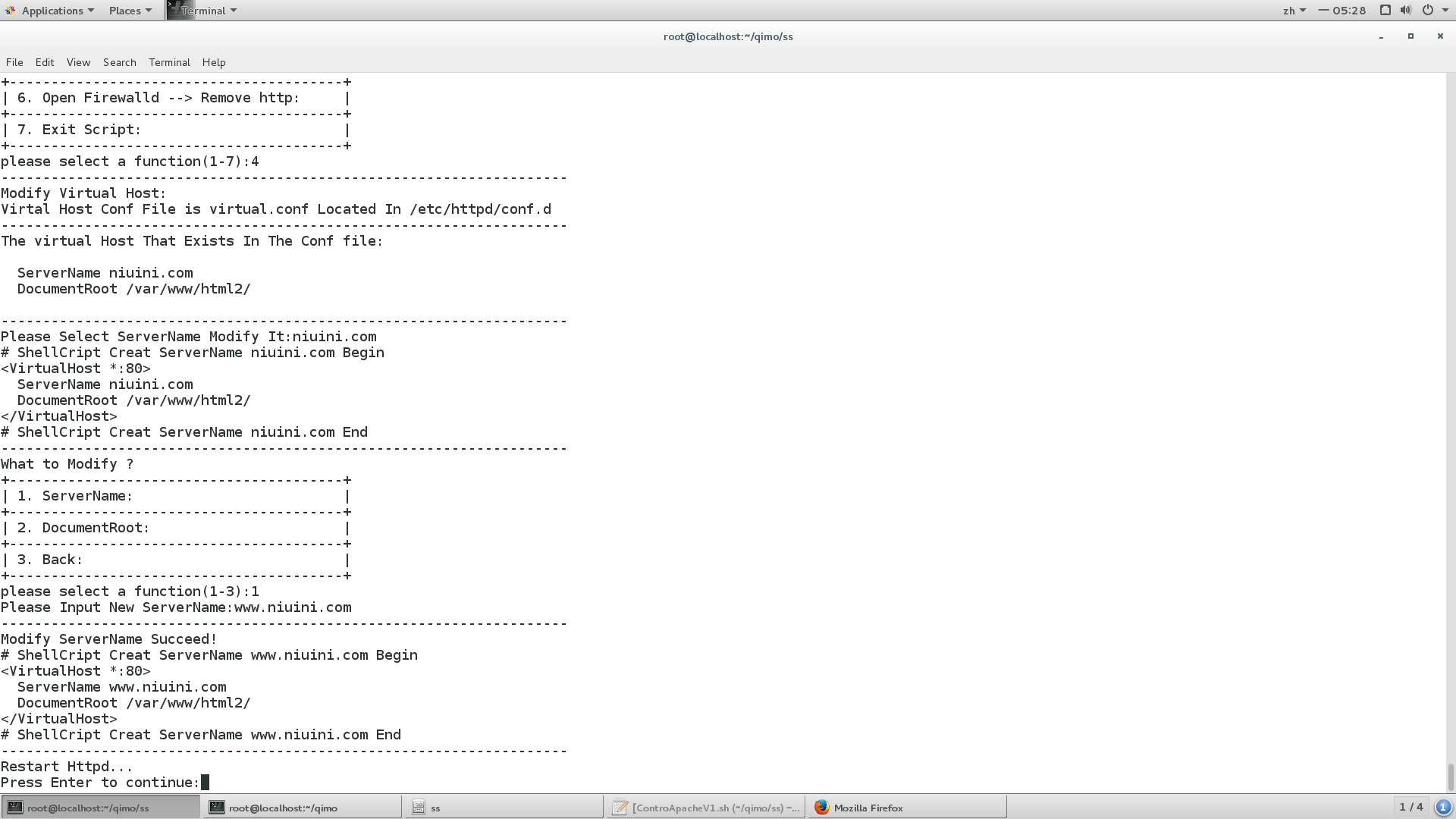
### 修改虚拟主机

输入需要修改的主机域名

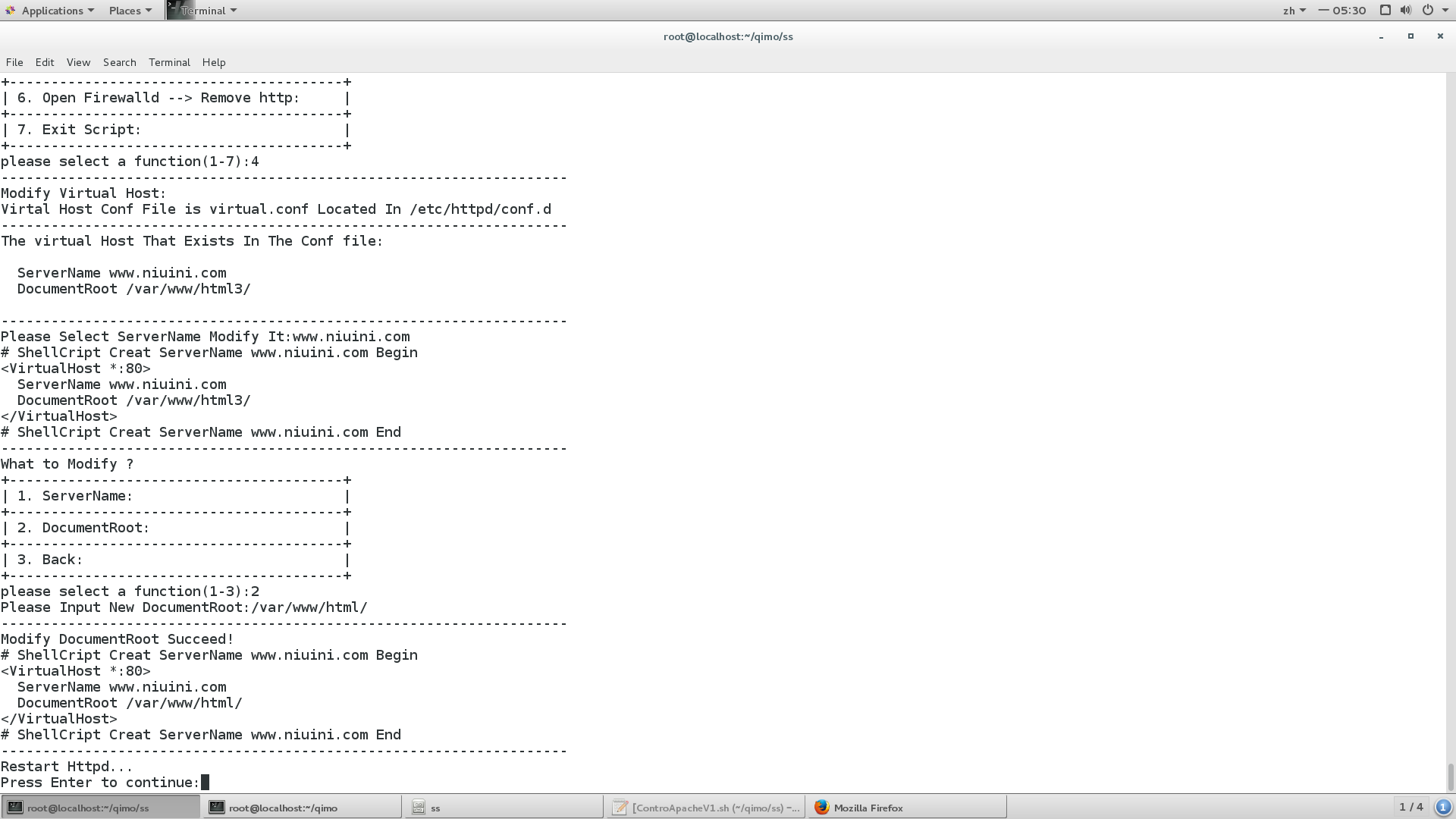
****选择修改域名还是文档目录

****

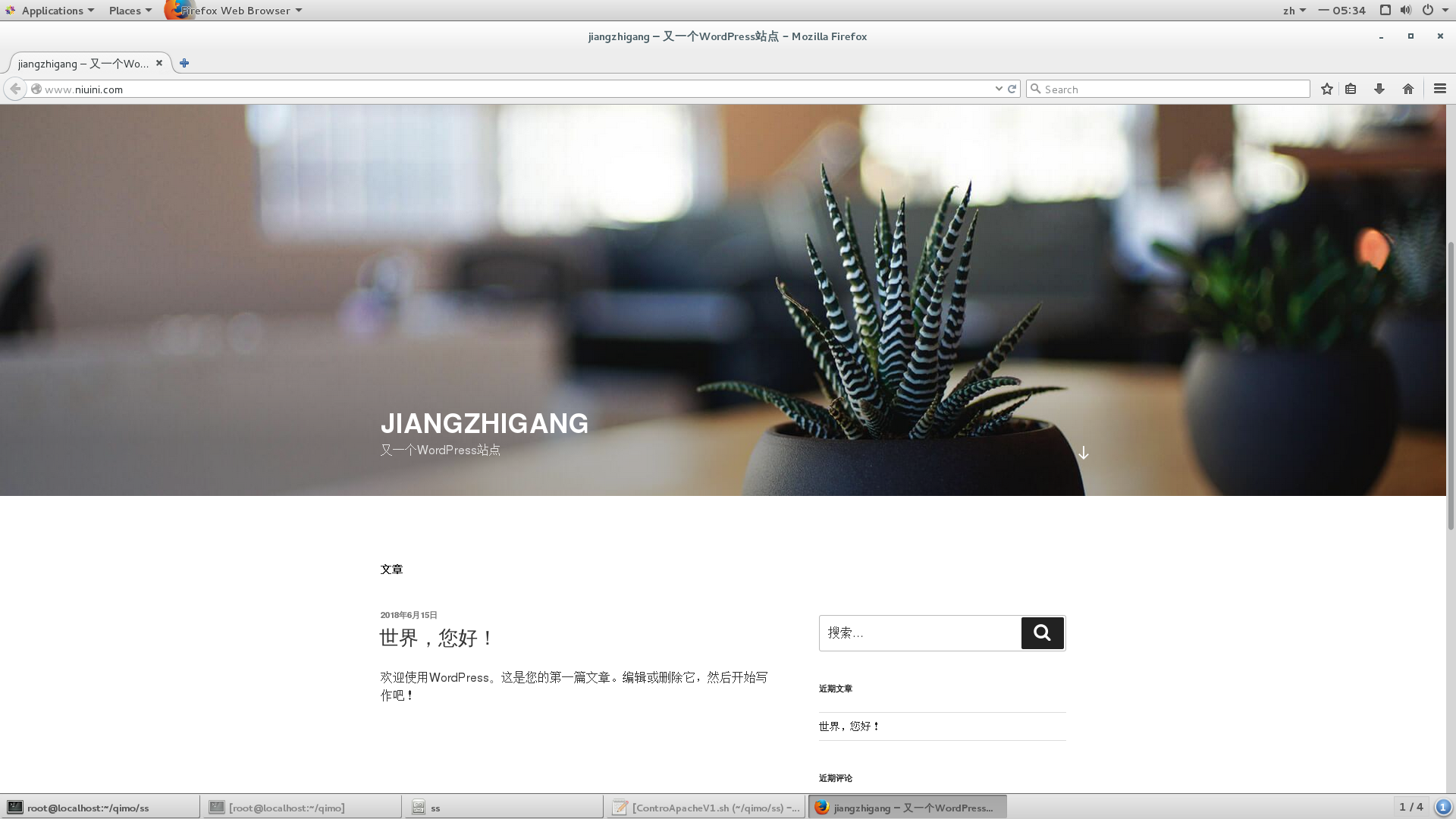
修改域名成功修改为[www.niuini.com](http://www.niuini.com)

****

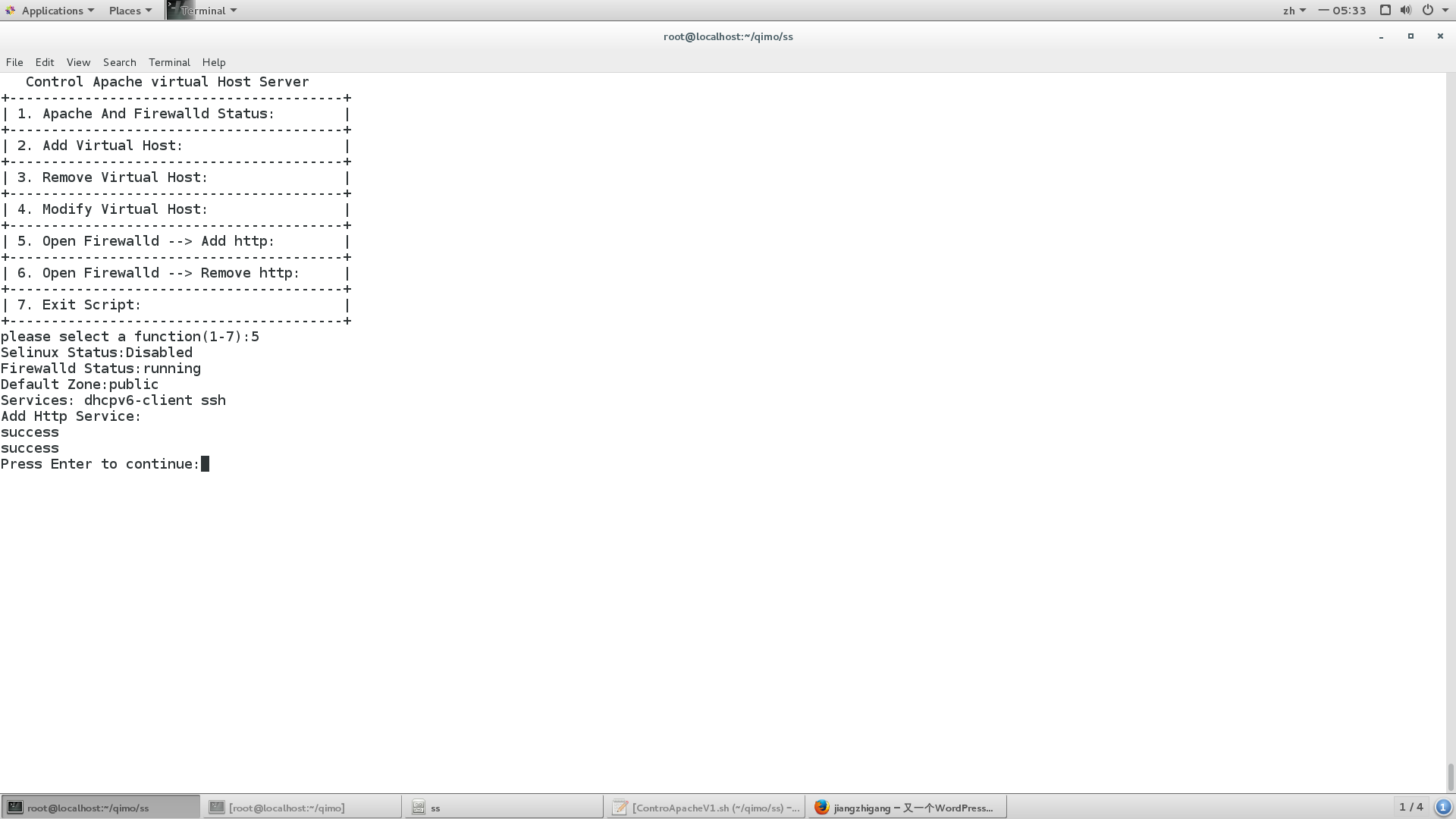
修改文档目录为 /var/www/html/

****

访问成功

****

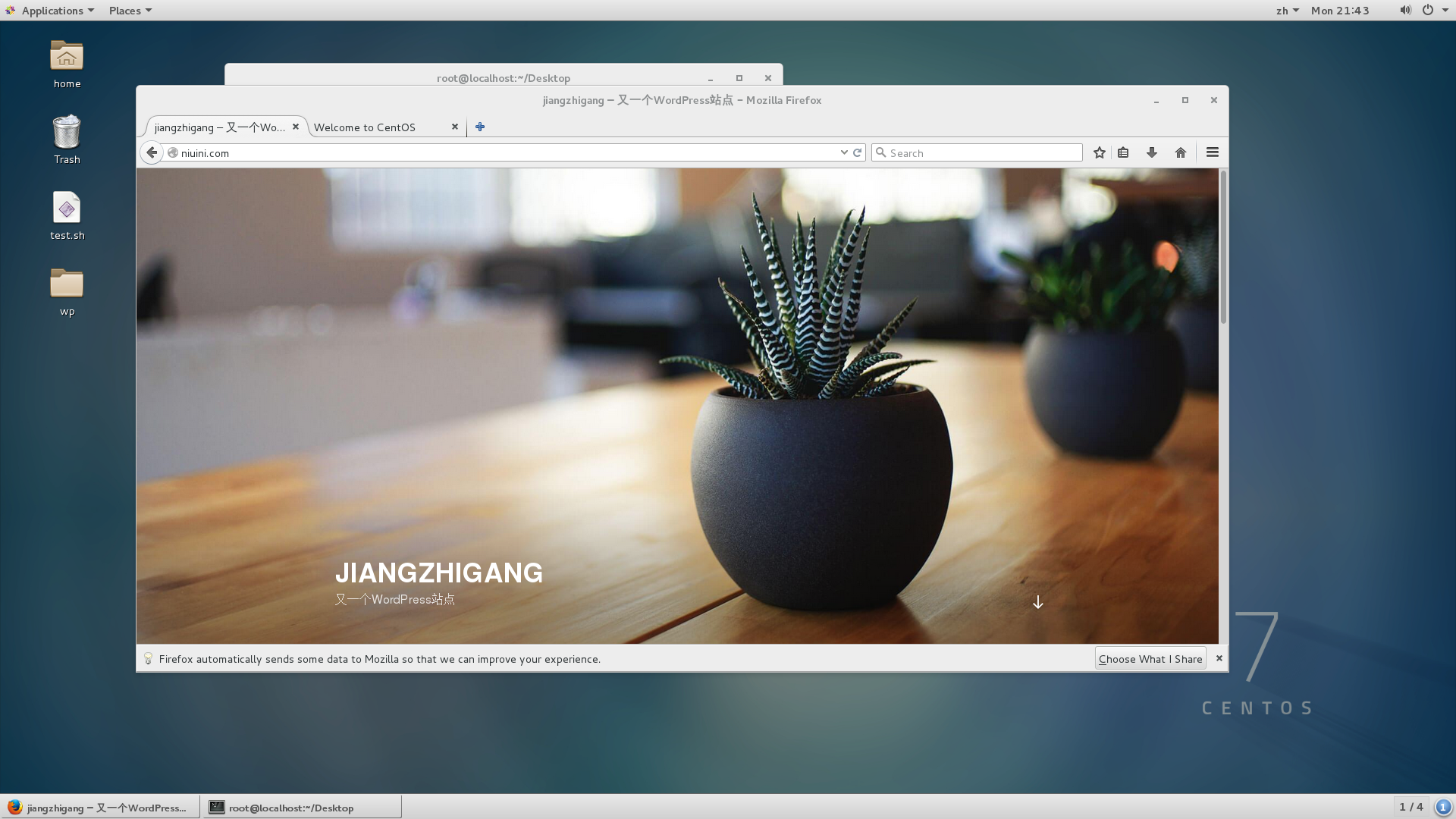
### 打开防火墙并且添加http服务永久生效

****

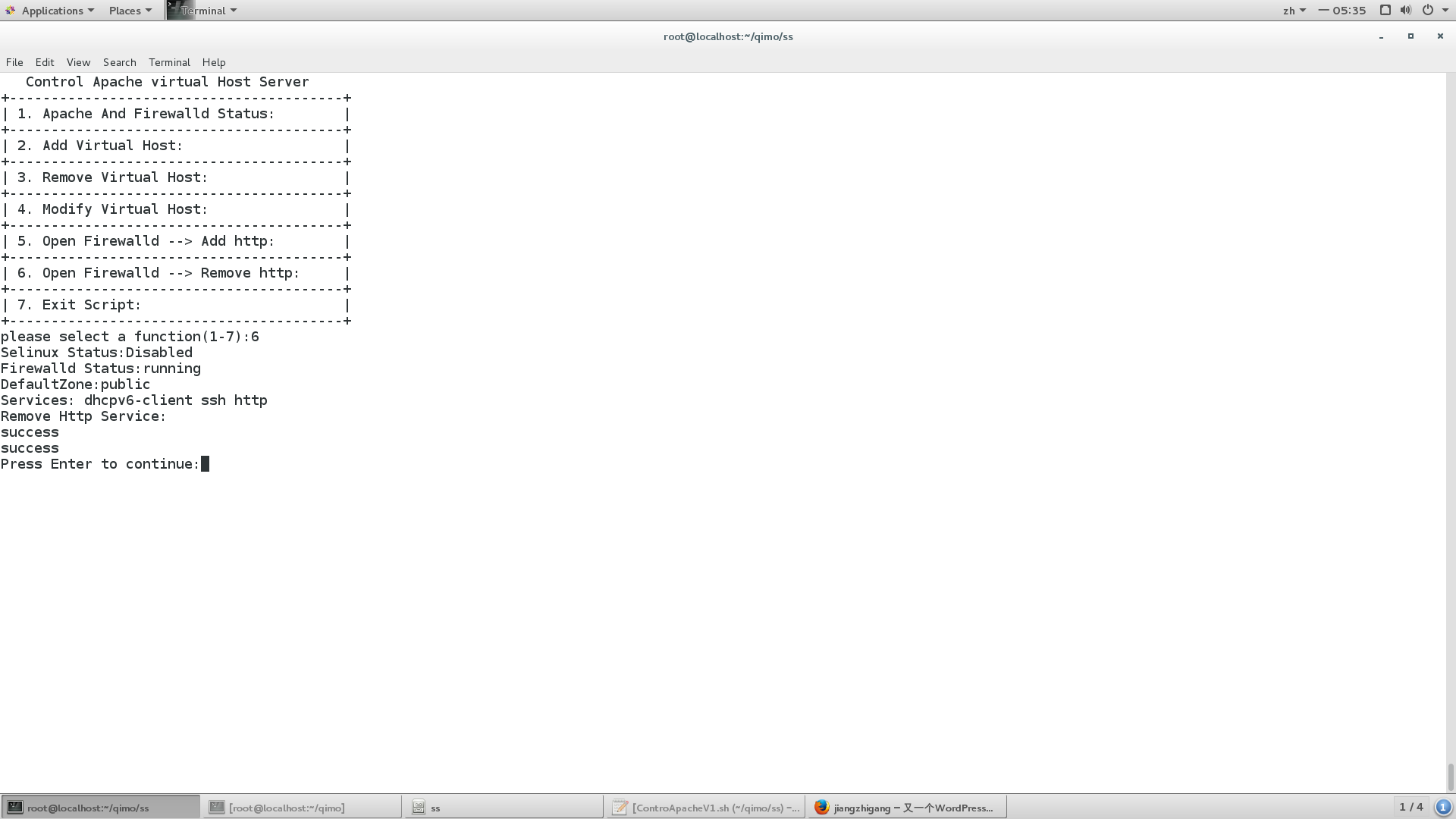
如果已经添加http服务则提示

****

客户端 访问

****

### 打开防火墙并且移除http服务永久生效

****

客户端访问

****

### 退出脚本

## 总结与展望

### 总结：

1.使用3台虚拟主机，搭建了一个小型网络，dhcp为其提供ip，dns为其提供域名解析，apache提供网页服务。

2.Dns和dhcp分开搭建，这样dhcp可以分配固定ip给dns服务器。

3.Dns客户端修改/ etc/resolve.conf 重启后失效 可以修改对应网卡配置文件

vim /etc/sysconfig/network-scripts/ifcfg-eno16777736

添加

NM\_CONTROLLED=no

PEERDNS=yes

DNS1=192.168.140.102

这样重启后会自动添加 nameserver 192.168.140.102 到/ etc/resolve.conf

4. 配置apache基于域名的虚拟主机，应配置dns，使多个域名指向相同apache服务器，当apache服务器接受浏览器发送的不同头文件，其中包括用户输入的域名，来选择虚拟主机，指向不同目录，从而完成1台web服务器绑定多个域名。当客服端浏览器输入不在虚拟主机目录中的域名，而该域名又指向apache服务器，测试显示：会访问第一个虚拟主机的文档目录作为默认目录，而不会访问httpd.conf中的DocumentRoot。因为设置了NameVirtualHost \*:80 。

### 展望：

虚拟主机的使用越来越频繁，一台主机可以绑定多个域名，符合当前一些用户的需求，该脚本功能比较简单，但是用户体验较好，会检测用户的输入，当用户输入为空，或者输入不存在，或者重复的内容时会提示。以后进一步完善脚本功能。

## 参考文献

1.周奇. Linux网络服务器配置、管理与实践教程[M]. 北京:清华大学出版社, 2011. 188-191

2. <https://blog.csdn.net/u011304615/article/details/71450847>； sed 匹配内容的前一行和后一行　添加内容，；2018年6月25日引自csdn博客

3. <https://www.cnblogs.com/wangkongming/p/3684905.html>； linux利用grep查看打印匹配的下几行或前后几行的命令 ；2018年6月25日引自cnblogs博客

4. https://blog.csdn.net/yf210yf/article/details/9207147；shell脚本----if（数字条件，字符串条件，字符串为空；2018年6月25日引自csdn博客

5. https://www.cnblogs.com/zqifa/p/linux-firewall-1.html；centos7下Firewall使用详解；2018年6月25日引自cnblogs博客