

# NSD OPERATION DAY07

- 1. [案例1：配置GRE VPN](#)
- 2. [案例2：创建PPTP VPN](#)
- 3. [案例3：创建L2TP+IPSec VPN](#)
- 4. [案例4：NTP时间同步](#)

## 1 案例1：配置GRE VPN

### 1.1 问题

本案例要求搭建一个GRE VPN环境，并测试该VPN网络是否能够正常通讯，要求如下：

- 启用内核模块ip\_gre
- 创建一个虚拟VPN隧道(10.10.10.0/24)
- 实现两台主机点到点的隧道通讯

### 1.2 方案

使用lsmod查看当前计算机已经加载的模块，使用modprobe加载Linux内核模块，使用modinfo可以查看内核模块的信息。

准备实验所需的虚拟机环境，实验环境所需要的主机及对应的IP设置列表如表-1所示，正确配置IP地址、主机名称，并且为每台主机配置YUM源。

表 - 1 主机列表

主机名	IP 地址
client	eth3(201.1.2.10/24)
proxy	eth0(192.168.4.5/24) eth3(201.1.2.5/24)

实验拓扑如图-1所示。

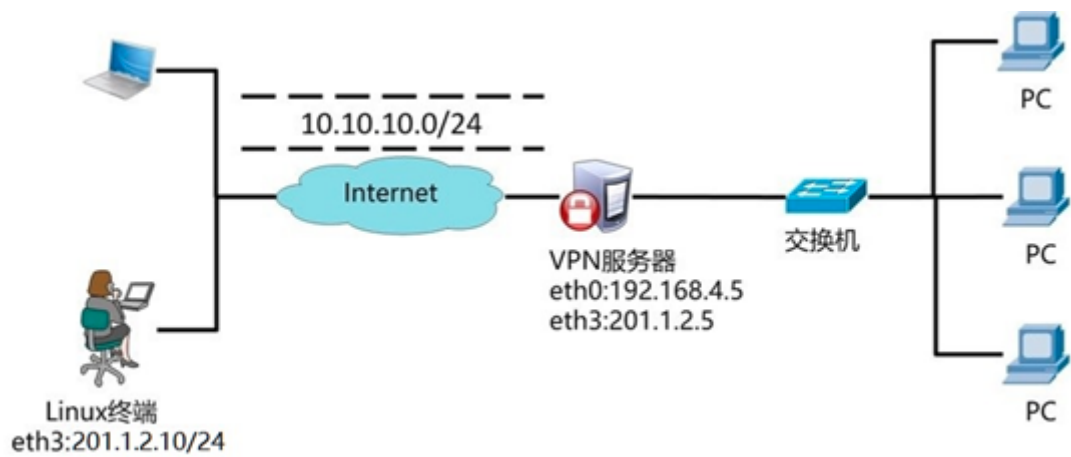


图-1

### 1.3 步骤

实现此案例需要按照如下步骤进行。

**步骤一：启用GRE模块 (client和proxy都需要操作)**

[Top](#)

### 1) 查看计算机当前加载的模块

```
01. [root@client ~]# lsmod //显示模块列表
02. [root@client ~]# lsmod | grep ip_gre //确定是否加载了gre模块
```

### 2) 加载模块ip\_gre

```
01. [root@client ~]# modprobe ip_gre
```

### 3) 查看模块信息

```
01. [root@client ~]# modinfo ip_gre
02. filename: /lib/modules/3.10.0-693.el7.x86_64/kernel/net/ipv4/ip_gre.ko.xz
03. alias: netdev-gretap0
04. alias: netdev-gre0
05. alias: rtnl-link-gretap
06. alias: rtnl-link-gre
07. license: GPL
08. rhelversion: 7.4
09. srcversion: F37A2BF90692F86E3A8BD15
10. depends: ip_tunnel,gre
11. intree: Y
12. vermagic: 3.10.0-693.el7.x86_64 SMP mod_unload modversions
13. signer: CentOS Linux kernel signing key
14. sig_key: DA:18:7D:CA:7D:BE:53:AB:05:BD:13:BD:0C:4E:21:F4:22:B6:A4:9C
15. sig_hashalgo: sha256
16. parm: log_ecn_error:Log packets received with corrupted ECN (bool)
17.
```

## 步骤二：Client主机创建VPN隧道

### 1) 创建隧道

```
01. [root@client ~]# ip tunnel add tun0 mode gre \
02. > remote 201.1.2.5 local 201.1.2.10
03. //ip tunnel add创建隧道（隧道名称为tun0），ip tunnel help可以查看帮助Top
04. //mode设置隧道使用gre模式
05. //local后面跟本机的IP地址，remote后面是与其他主机建立隧道的对方IP地址
```

## 2) 启用该隧道（类似与设置网卡up）

```
01. [root@client ~]# ip link show
02. [root@client ~]# ip link set tun0 up //设置UP
03. [root@client ~]# ip link show
```

## 2) 为VPN配置隧道IP地址

```
01. [root@client ~]# ip addr add 10.10.10.10/24 peer 10.10.10.5/24 \
02. > dev tun0
03. //为隧道tun0设置本地IP地址（10.10.10.10/24）
04. //隧道对面的主机IP的隧道IP为10.10.10.5/24
05. [root@client ~]# ip a s //查看IP地址
```

## 3) 关闭防火墙

```
01. [root@client ~]# firewall-cmd --set-default-zone=trusted
```

## 步骤三：Proxy主机创建VPN隧道

### 1) 查看计算机当前加载的模块

```
01. [root@client ~]# lsmod //显示模块列表
02. [root@client ~]# lsmod | grep ip_gre //确定是否加载了gre模块
```

### 2) 加载模块ip\_gre

```
01. [root@client ~]# modprobe ip_gre
```

### 3) 创建隧道

```
01. [root@proxy ~]# ~]# ip tunnel add tun0 mode gre \
02. > remote 201.1.2.10 local 201.1.2.5
03. //ip tunnel add创建隧道（隧道名称为tun0），ip tunnel help可以查看帮助
```

[Top](#)

04. //mode设置隧道使用gre模式
05. //local后面跟本机的IP地址，remote后面是与其他主机建立隧道的对方IP地址

#### 4) 启用该隧道（类似与设置网卡up）

01. [root@proxy ~]# ip link show
02. [root@proxy ~]# ip link set tun0 up //设置UP
03. [root@proxy ~]# ip link show

#### 5) 为VPN配置隧道IP地址

01. [root@proxy ~]# ip addr add 10.10.10.5/24 peer 10.10.10.10/24 \
02. > dev tun0
03. //为隧道tun0设置本地IP地址（10.10.10.5/24）
04. //隧道对面的主机IP的隧道IP为10.10.10.10/24
05. [root@proxy ~]# ip a s //查看IP地址

#### 6) 开启路由转发、关闭防火墙

01. [root@proxy ~]# echo "1" > /proc/sys/net/ipv4/ip\_forward
02. [root@proxy ~]# firewall-cmd --set-default-zone=trusted

#### 7)测试连通性

01. [root@client ~]# ping 10.10.10.5
02. [root@proxy ~]# ping 10.10.10.10

## 2 案例2：创建PPTP VPN

### 2.1 问题

本案例要求搭建一个PPTP VPN环境，并测试该VPN网络是否能够正常通讯，要求如下：

- 使用PPTP协议创建一个支持身份验证的隧道连接
- 使用MPPE对数据进行加密
- 为客户端分配192.168.3.0/24的地址池
- 客户端连接的用户名为jacob，密码为123456

[Top](#)

### 2.2 方案

准备实验所需的虚拟机环境，实验环境所需要的主机及对应的IP设置列表如表-2所示，正确配置IP地址、主机名称，并且为每台主机配置YUM源。

表 - 2 主机列表

主机名	IP 地址
windows 主机	网卡桥接 public2(201.1.2.20/24)
proxy	eth0(192.168.4.5/24) eth3(201.1.2.5/24)

实验拓扑如图-2所示。

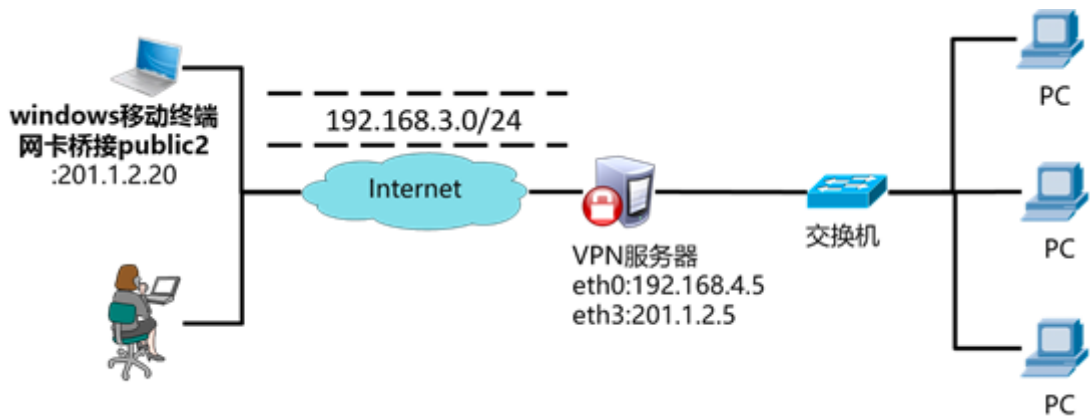


图-2

## 2.3 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：部署VPN服务器

1) 安装软件包（软件包参考lnmp\_soft）

01. [root@proxy ~]# yum localinstall pptpd-1.4.0-2.el7.x86\_64.rpm
02. [root@proxy ~]# rpm -qc pptpd
03. /etc/ppp/options.pptpd
04. /etc/pptpd.conf
05. /etc/sysconfig/pptpd

2)修改配置文件

01. [root@proxy ~]# vim /etc/pptpd.conf
02. .. ..
03. localip 201.1.2.5 //服务器本地IP
04. remoteip 192.168.3.1-50 //分配给客户端的IP池
- 05.
06. [root@proxy ~]# vim /etc/ppp/options.pptpd

[Top](#)

```

07. require-mppe-128 //使用MPPE加密数据
08. ms-dns 8.8.8.8 //DNS服务器
09.
10. [root@proxy ~]# vim /etc/ppp/chap-secrets //修改账户配置文件
11. jacob * 123456 *
12. //用户名 服务器标记 密码 客户端
13.
14. [root@proxy ~]# echo "1" > /proc/sys/net/ipv4/ip_forward //开启路由转发

```

### 3) 启动服务

```

01. [root@proxy ~]# systemctl start pptpd
02. [root@proxy ~]# systemctl enable pptpd
03. [root@proxy ~]# firewall-cmd --set-default-zone=trusted

```

### 4) 翻墙设置（非必需操作）

```

01. [root@proxy ~]# iptables -t nat -A POSTROUTING -s 192.168.3.0/24 \
02. > -j SNAT --to-source 201.1.2.5

```

## 步骤二：客户端设置

启动一台Windows虚拟机，将虚拟机网卡桥接到public2，配置IP地址为201.1.2.20。

新建网络连接（具体操作如图-3所示），输入VPN服务器账户与密码（具体操作如图-4所示），连接VPN并测试网络连通性（如图-5所示）。



图-3



图-4

```
C:\Users\Jacob>ping 201. 1. 2. 5
```

```
C:\Users\Jacob>ping 192. 168. 4. 5
```

图-5

### 3 案例3：创建L2TP+IPSec VPN

#### 3.1 问题

本案例要求搭建一个L2TP+IPSec VPN环境，并测试该VPN网络是否能够正常通讯，具体要求如下：

- 使用L2TP协议创建一个支持身份验证与加密的隧道连接
- 使用IPSec对数据进行加密
- 为客户端分配192.168.3.0/24的地址池
- 客户端连接的用户名为：jacob，密码为：123456
- 预共享密钥为：randpass

#### 3.2 方案

准备实验所需的虚拟机环境，实验环境所需要的主机及对应的IP设置列表如表-3所示，正确配置IP地址、主机名称，并且为每台主机配置YUM源。

表 - 3 主机列表

主机名	IP 地址
windows 主机	网卡桥接 public2(201.1.2.20/24)
client(作为 vpn 服务器)	eth0(192.168.4.10/24) eth3(201.1.2.10/24)

实验拓扑如图-6所示。

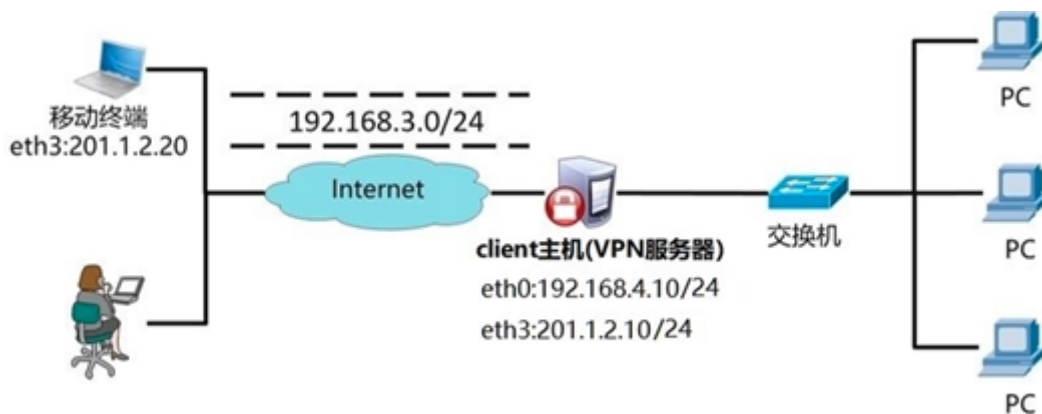


图-6

### 3.3 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：部署IPSec服务

##### 1) 安装软件包

```
01. [root@client ~]# yum -y install libreswan
```

##### 2) 新建IPSec密钥验证配置文件

```
01. [root@client ~]# cat /etc/ipsec.conf           //仅查看一下该主配置文件
02. ...
03. include /etc/ipsec.d/*.conf                   //加载该目录下的所有配置文件
04.
05. [root@client ~]# vim /etc/ipsec.d/myipsec.conf
06. //新建该文件，参考lnmp_soft/vpn/myipsec.conf
07. conn IDC-PSK-NAT
08.     rightsubnet=vhost:%priv                     //允许建立的VPN虚拟网络
09.     also=IDC-PSK-noNAT
10.
11. conn IDC-PSK-noNAT
12.     authby=secret                               //加密认证
13.     ike=3des-sha1;modp1024                       //算法
14.     phase2alg=aes256-sha1;modp2048               //算法
15.     pfs=no
16.     auto=add
17.     keyingtries=3
18.     rekey=no
19.     ikelifetime=8h
```

[Top](#)



```

20.    keylife=3h
21.    type=transport
22.    left=201.1.2.10           //重要，服务器本机的外网IP
23.    leftprotoport=17/1701
24.    right=%any               //允许任何客户端连接
25.    rightprotoport=17/%any

```

### 3)创建IPSec预定义共享密钥

```

01.    [root@client ~]# cat /etc/ipsec.secrets           //仅查看，不要修改该文件
02.    include /etc/ipsec.d/*.secrets
03.
04.    [root@client ~]# vim /etc/ipsec.d/mypass.secrets   //新建该文件
05.    201.1.2.10 %any: PSK "randpass"                 //randpass为预共享密钥
06.                                                    //201.1.2.10是VPN服务器的IP

```

### 4)启动IPSec服务

```

01.    [root@client ~]# systemctl start ipsec
02.    [root@client ~]# netstat -ntulp |grep pluto
03.    udp      0      0 127.0.0.1:4500      0.0.0.0:*        3148/pluto
04.    udp      0      0 192.168.4.10:4500   0.0.0.0:*        3148/pluto
05.    udp      0      0 201.1.2.10:4500     0.0.0.0:*        3148/pluto
06.    udp      0      0 127.0.0.1:500       0.0.0.0:*        3148/pluto
07.    udp      0      0 192.168.4.10:500    0.0.0.0:*        3148/pluto
08.    udp      0      0 201.1.2.10:500      0.0.0.0:*        3148/pluto
09.    udp6     0      0 :::1:500            :::*              3148/pluto

```

## 步骤二：部署XL2TP服务

### 1) 安装软件包（软件包参考lnmp\_soft）

```

01.    [root@client ~]# yum localinstall xl2tpd-1.3.8-2.el7.x86_64.rpm

```

### 2) 修改xl2tp配置文件（修改3个配置文件的内容）

[Top](#)

```

01.    [root@client ~]# vim /etc/xl2tpd/xl2tpd.conf     //修改主配置文件

```

```

02. [global]
03. .. ..
04. [lns default]
05. .. ..
06. ip range = 192.168.3.128-192.168.3.254 //分配给客户端的IP池
07. local ip = 201.1.2.10 //VPN服务器的IP地址
08.
09. [root@client ~]# vim /etc/ppp/options.xl2tpd //认证配置
10. require-mschap-v2 //添加一行，强制要求认证
11. #crtsects //注释或删除该行
12. #lock //注释或删除该行
13.
14. root@client ~]# vim /etc/ppp/chap-secrets //修改密码文件
15. jacob * 123456 * //账户名称 服务器标记 密码 客户端IP

```

### 3) 启动服务

```

01. [root@client ~]# systemctl start xl2tpd
02. [root@client ~]# netstat -ntulp |grep xl2tpd
03. udp 0 0 0.0.0.0:1701 0.0.0.0:* 3580/xl2tpd

```

### 4) 设置路由转发，防火墙

```

01. [root@client ~]# echo "1" > /proc/sys/net/ipv4/ip_forward
02. [root@client ~]# firewall-cmd --set-default-zone=trusted

```

### 5) 翻墙设置（非必需操作）

```

01. [root@client ~]# iptables -t nat -A POSTROUTING -s 192.168.3.0/24 \
02. > -j SNAT --to-source 201.1.2.10

```

## 步骤二：客户端设置

启动一台Windows虚拟机，将虚拟机网卡桥接到public2，配置IP地址为201.1.2.20。

1. 新建网络连接（参考案例2），输入VPN服务器账户与密码（参考案例2）。

设置VPN连接的属性，预共享密钥是IPSec配置文件中填写的randpass，具体操作如图1-10所示。

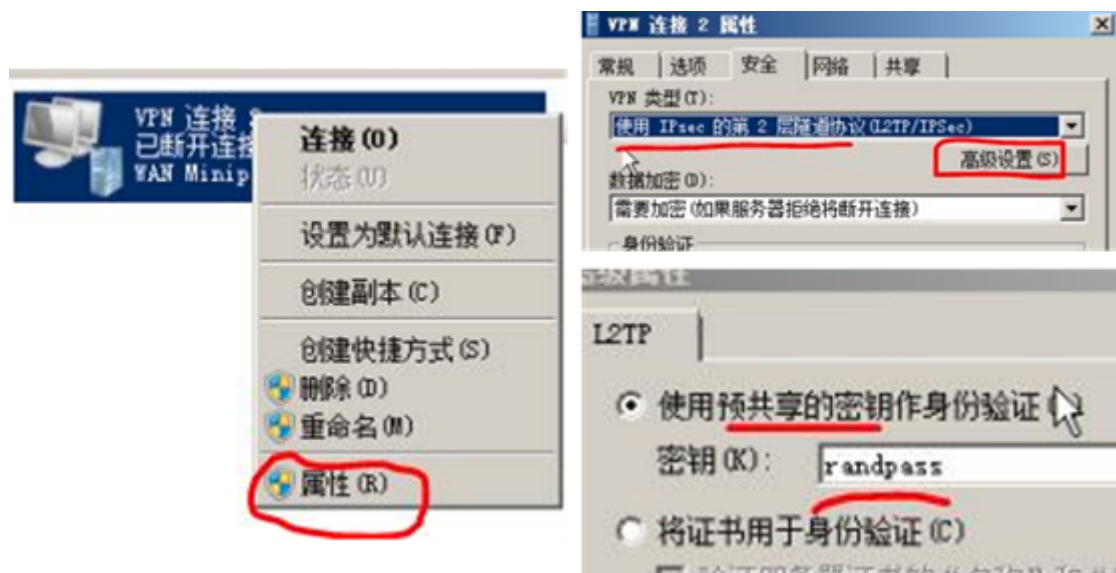


图-7

2. 设置Windows注册表（不修改注册表，连接VPN默认会报789错误），具体操作如下：

- 单击"开始"，单击"运行"，键入"regedit"，然后单击"确定"
- 找到下面的注册表子项，然后单击它：
- HKEY\_LOCAL\_MACHINE\ System\CurrentControlSet\Services\Rasman\Parameters
- 在"编辑"菜单上，单击"新建"->"DWORD值"
- 在"名称"框中，键入"ProhibitIpSec"
- 在"数值数据"框中，键入"1"，然后单击"确定"
- 退出注册表编辑器，然后重新启动计算机

连接VPN并测试网络连通性（参考案例2）。

## 4 案例4：NTP时间同步

### 4.1 问题

本案例要求搭建一个NTP服务器，为整个网络环境中的所有主机提供时间校准服务，具体要求如下：

- 部署一台NTP时间服务器
- 设置时间服务器上层与0.centos.pool.ntp.org同步
- 设置本地服务器层级数量为10
- 允许192.168.4.0/24网络的主机同步时间
- 客户端验证时间是否同步

### 4.2 方案

准备实验所需的虚拟机环境，实验环境所需要的主机及对应的IP设置列表如表-4所示，正确配置IP地址、主机名称，并且为每台主机配置YUM源。

表 - 4 主机列表

主机名	IP 地址
client	eth0 (192.168.4.10/24)
proxy	eth0(192.168.4.5/24)
	eth1(192.168.2.5/24)

[Top](#)

实验拓扑如图-8所示。

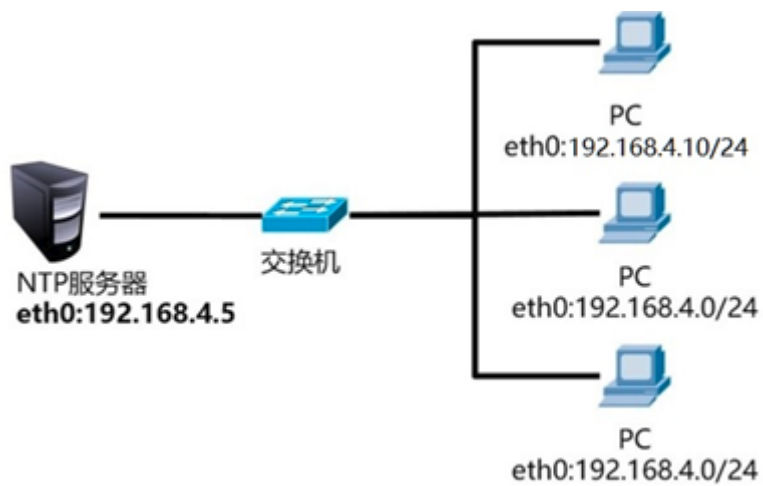


图-8

Network Time Protocol（网络时间协议）采用的是分层设计，如图-9所示，Stratum层的总数限制在15以内（包括15）。

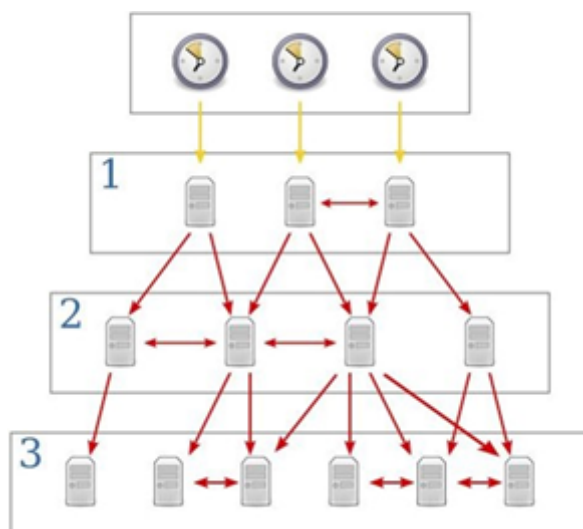


图-9

## 4.3 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：部署NTP服务

#### 1) 安装软件包

01. `[root@proxy ~]# yum -y install chrony`
02. `[root@proxy ~]# rpm -qc chrony` //查看配置文件列表
03. `/etc/chrony.conf`
04. `/etc/chrony.keys`
05. `.. ..`

[Top](#)

#### 2)修改配置文件

```
01. [root@proxy ~]# cat /etc/chrony.conf
02. .. ..
03. server 0.centos.pool.ntp.org iburst //server用户客户端指向上层NTP服务器
04. allow 192.168.4.0/24 //允许那个IP或网络访问NTP
05. #deny 192.168.4.1 //拒绝那个IP或网络访问NTP
06. local stratum 10 //设置NTP服务器的层数量
07. .. ..
```

#### 4)启动NTP服务

```
01. [root@proxy ~]# systemctl restart chronyd
02. [root@proxy ~]# systemctl enable chronyd
```

#### 5)设置防火墙

```
01. [root@proxy ~]# firewall-cmd --set-default-zone=trusted
```

### 步骤二：配置客户端

#### 1) 安装软件包

```
01. [root@client ~]# yum -y install chrony
```

#### 2) 修改配置文件

```
01. [root@client ~]# vim /etc/chrony.conf
02. server 192.168.4.5 iburst //设置与哪台服务器同步数据
03. //iburst参数设置重启服务后尽快同步时间
```

#### 3) 将客户端时间修改为错误的时间

```
01. [root@client ~]# date -s "hour:minute" //调整时间（小时：分钟）
02. [root@client ~]# date //查看修改后的时间
```

[Top](#)

#### 4) 重启chrony与服务器同步时间

```
01. [root@client ~]# systemctl restart chronyd
```

## 5) 确认时间是否已经同步

```
01. [root@client ~]# date //多执行几次查看结果
```