# NSD OPERATION DAY07

1. [案例1：配置GRE VPN](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201801/OPERATION/DAY07/CASE/01/index.html" \l "case1)
2. [案例2：创建PPTP VPN](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201801/OPERATION/DAY07/CASE/01/index.html" \l "case2)
3. [案例3：创建L2TP+IPSec VPN](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201801/OPERATION/DAY07/CASE/01/index.html" \l "case3)
4. [案例4：NTP时间同步](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201801/OPERATION/DAY07/CASE/01/index.html" \l "case4)
5. [案例5：pssh远程套件工具](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201801/OPERATION/DAY07/CASE/01/index.html" \l "case5)

## **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



实验拓扑如图-1所示。

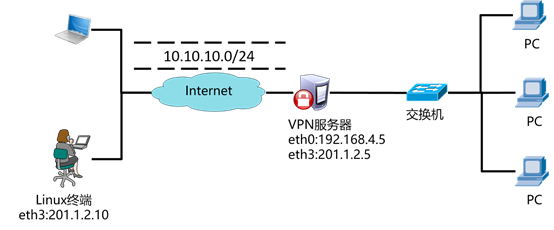


图-1

### **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，下面以client为例）**

1）查看计算机当前加载的模块

1. **[**root@client **~]**# lsmod                            //显示模块列表
2. **[**root@client **~]**# lsmod **|** grep ip\_gre            //确定是否加载了gre模块

2)加载模块ip\_gre

1. **[**root@client **~]**# modprobe ip\_gre

3）查看模块信息

1. **[**root@client **~]**# modinfo ip\_gre
2. filename**:** /lib/modules**/3.10.0-693.**el7**.**x86\_64**/**kernel**/**net**/**ipv4**/**ip\_gre**.**ko**.**xz
3. alias**:** netdev**-**gretap0
4. alias**:** netdev**-**gre0
5. alias**:** rtnl**-**link**-**gretap
6. alias**:** rtnl**-**link**-**gre
7. license**:** GPL
8. rhelversion**:** **7.4**
9. 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**)**

**步骤二：Client主机创建VPN隧道**

1）创建隧道

1. **[**root@client **~]**# ip tunnel add tun0 mode gre **\**
2. **>** remote **201.1.2.5** local **201.1.2.10**
3. //ip tunnel add创建隧道（隧道名称为tun0），ip tunnel help可以查看帮助
4. //mode设置隧道使用gre模式
5. //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）

1. **[**root@client **~]**# ip link show
2. **[**root@client **~]**# ip link set tun0 up         //设置UP
3. **[**root@client **~]**# ip link show

2）为VPN配置隧道IP地址

1. **[**root@client **~]**# ip addr add **10.10.10.10/24** peer **10.10.10.5/24** **\**
2. **>** dev tun0
3. //为隧道tun0设置本地IP地址（10.10.10.10.10/24）
4. //隧道对面的主机IP的隧道IP为10.10.10.5/24
5. **[**root@client **~]**# ip a s //查看IP地址

3）关闭防火墙

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

**步骤三：Proxy主机创建VPN隧道**

1）创建隧道

1. **[**root@proxy **~]**# **~]**# ip tunnel add tun0 mode gre **\**
2. **>** remote **201.1.2.10** local **201.1.2.5**
3. //ip tunnel add创建隧道（隧道名称为tun0），ip tunnel help可以查看帮助
4. //mode设置隧道使用gre模式
5. //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）

1. **[**root@proxy **~]**# ip link show
2. **[**root@proxy **~]**# ip link set tun0 up         //设置UP
3. **[**root@proxy **~]**# ip link show

2）为VPN配置隧道IP地址

1. **[**root@proxy **~]**# ip addr add **10.10.10.5/24** peer **10.10.10.10/24** **\**
2. **>** dev tun0
3. //为隧道tun0设置本地IP地址（10.10.10.10.5/24）
4. //隧道对面的主机IP的隧道IP为10.10.10.10/24
5. **[**root@proxy **~]**# ip a s //查看IP地址

3）开启路由转发、关闭防火墙

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

4)测试连通性

1. **[**root@client **~]**# ping **10.10.10.5**
2. **[**root@proxy **~]**# ping **10.10.10.10**

## **2 案例2：创建PPTP VPN**

### **2.1 问题**

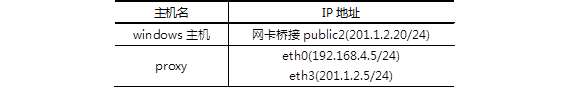
本案例要求搭建一个PPTP VPN环境，并测试该VPN网络是否能够正常通讯，要求如下:

* 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

### **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



实验拓扑如图-2所示。

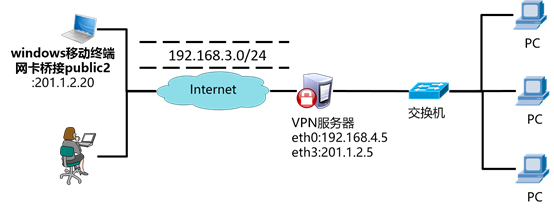


图-2

### **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）

1. **[**root@proxy **~]**# yum localinstall pptpd**-1.4.0-2.**el7**.**x86\_64**.**rpm
2. **[**root@proxy **~]**# rpm **-**qc pptpd
3. /etc/ppp**/**options**.**pptpd
4. /etc/pptpd**.**conf
5. /etc/sysconfig**/**pptpd

2)修改配置文件

1. **[**root@proxy **~]**# vim **/**etc**/**pptpd**.**conf
2. **..** **..**
3. localip **201.1.2.5**                                    //服务器本地IP
4. remoteip **192.168.3.1-50**                            //分配给客户端的IP池
5. **[**root@proxy **~]**# vim **/**etc**/**ppp**/**options**.**pptpd
6. require**-**mppe**-128**                                    //使用MPPE加密数据
7. ms**-**dns **8.8.8.8**                                    //DNS服务器
8. **[**root@proxy **~]**# vim **/**etc**/**ppp**/**chap**-**secrets            //修改账户配置文件
9. jacob         **\***          **123456**     **\***
10. //用户名    服务器标记    密码    客户端
11. **[**root@proxy **~]**# echo "1" **>** /proc/sys**/**net**/**ipv4**/**ip\_forward    //开启路由转发

3）启动服务

1. **[**root@proxy **~]**# systemctl start pptpd
2. **[**root@proxy **~]**# systemctl enable pptpd
3. **[**root@proxy **~]**# firewall**-**cmd **--**set**-default-**zone**=**trusted

4）翻墙设置（非必需操作）

1. **[**root@proxy **~]**# iptables **-**t nat **-**A POSTROUTING **-**s **192.168.3.0/24** **\**
2. **>** **-**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）。

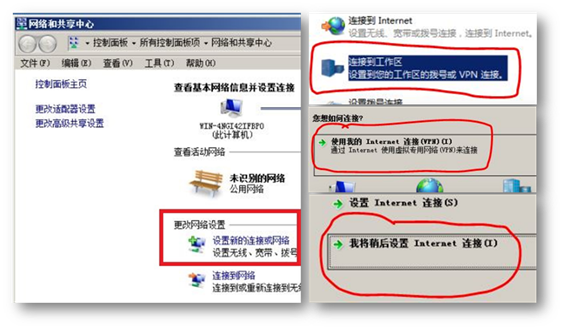


图-3



图-4

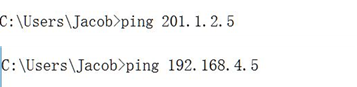


图-5

## **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



实验拓扑如图-6所示。

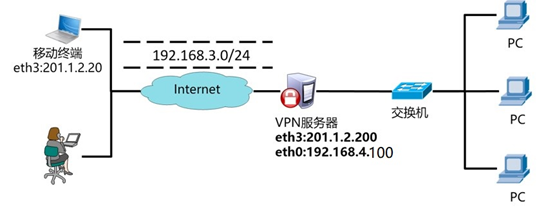


图-6

### **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

1. **[**root@client **~]**# yum **-**y install libreswan

2)新建IPSec密钥验证配置文件

1. **[**root@client **~]**# cat **/**etc**/**ipsec**.**conf                //仅查看一下该主配置文件
2. **..** **..**
3. include **/**etc**/**ipsec**.**d/\*.conf                    //加载该目录下的所有配置文件
4. [[root@client](mailto:root@client) ~]# vim /etc/ipsec.d/myipsec.conf
5. //新建该文件，参考lnmp\_soft/vpn/myipsec.conf
6. conn IDC-PSK-NAT
7. rightsubnet=vhost:%priv                        //允许建立的VPN虚拟网络
8. also=IDC-PSK-noNAT
9. conn IDC-PSK-noNAT
10. authby=secret                                    //加密认证
11. ike=3des-sha1;modp1024                        //算法
12. phase2alg=aes256-sha1;modp2048                //算法
13. pfs=no
14. auto=add
15. keyingtries=3
16. rekey=no
17. ikelifetime=8h
18. keylife=3h
19. type=transport
20. left=201.1.2.200                                //重要，服务器本机的外网IP
21. leftprotoport=17/1701
22. right=%any                                    //允许任何客户端连接
23. rightprotoport=17/%any

3)创建IPSec预定义共享密钥

1. **[**root@client **~]**# cat **/**etc**/**ipsec**.**secrets                 //仅查看，不要修改该文件
2. include **/**etc**/**ipsec**.**d/\*.secrets
3. [[root@client](mailto:root@client) ~]# vim /etc/ipsec.d/mypass.secrets        //新建该文件
4. 201.1.2.200 %any: PSK "randpass"             //randpass为密钥
5. //201.1.2.200是VPN服务器的IP

4)启动IPSec服务

1. **[**root@client **~]**# systemctl start ipsec
2. **[**root@client **~]**# netstat **-**ntulp **|**grep pluto
3. udp **0** **0** **127.0.0.1:4500** **0.0.0.0:\*** **3148/**pluto
4. udp **0** **0** **192.168.4.200:4500** **0.0.0.0:\*** **3148/**pluto
5. udp **0** **0** **201.1.2.200:4500** **0.0.0.0:\*** **3148/**pluto
6. udp **0** **0** **127.0.0.1:500** **0.0.0.0:\*** **3148/**pluto
7. udp **0** **0** **192.168.4.200:500** **0.0.0.0:\*** **3148/**pluto
8. udp **0** **0** **201.1.2.200:500** **0.0.0.0:\*** **3148/**pluto
9. 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）

1. **[**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）

1. **[**root@client **~]**# vim **/**etc**/**xl2tpd**/**xl2tpd**.**conf                //修改主配置文件
2. **[**global**]**
3. **..** **..**
4. **[**lns **default]**
5. **..** **..**
6. ip range **=** **192.168.3.128-192.168.3.254**                    //分配给客户端的IP池
7. local ip **=** **201.1.2.200**                                //VPN服务器的IP地址
8. **[**root@client **~]**# vim **/**etc**/**ppp**/**options**.**xl2tpd                //认证配置
9. require**-**mschap**-**v2                                         //添加一行，强制要求认证
10. #crtscts                                                //注释或删除该行
11. #lock                                                //注释或删除该行
12. root@client **~]**# vim **/**etc**/**ppp**/**chap**-**secrets                    //修改密码文件
13. jacob **\*** **123456** **\***                //账户名称 服务器标记 密码 客户端IP

3）启动服务

1. **[**root@client **~]**# systemctl start xl2tpd
2. **[**root@client **~]**# netstat **-**ntulp **|**grep xl2tpd
3. udp **0** **0** **0.0.0.0:1701** **0.0.0.0:\*** **3580/**xl2tpd

4）设置路由转发，防火墙

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

5）翻墙设置（非必需操作）

1. **[**root@client **~]**# iptables **-**t nat **-**A POSTROUTING **-**s **192.168.3.0/24** **\**
2. **>** **-**j SNAT **--**to**-**source **201.1.2.200**

**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-7所示。

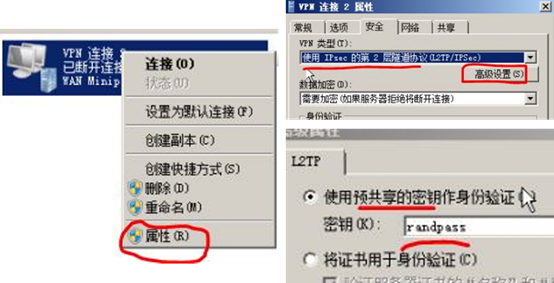


图-7

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**

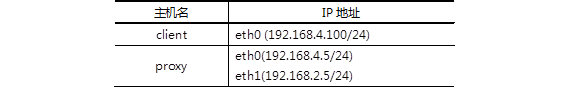
本案例要求搭建一个NTP服务器，为整个网络环境中的所有主机提供时间校准服务，具体要求如下：

* 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



实验拓扑如图-8所示。

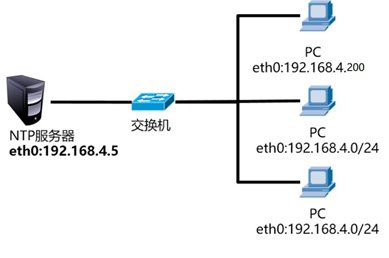


图-8

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）。

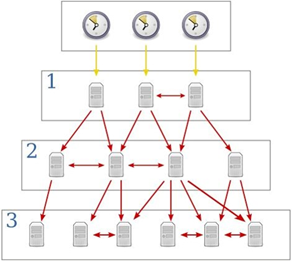


图-9

### **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

1. **[**root@proxy **~]**# yum **-**y install chrony
2. **[**root@proxy **~]**# rpm **-**qc chrony                        //查看配置文件列表
3. /etc/chrony**.**conf
4. /etc/chrony**.**keys
5. **..** **..**

2)修改配置文件

1. **[**root@proxy **~]**# cat **/**etc**/**chrony**.**conf
2. **..** **..**
3. server **0.**centos**.**pool**.**ntp**.**org iburst //server用户客户端指向上层NTP服务器
4. allow **192.168.4.0/24**                        //允许那个IP或网络访问NTP
5. #deny **192.168.4.1**                        //拒绝那个IP或网络访问NTP
6. local stratum **10**                            //设置NTP服务器的层数量
7. **..** **..**

4)启动NTP服务

1. **[**root@proxy **~]**# systemctl restart chronyd
2. **[**root@proxy **~]**# systemctl enable chronyd

5)设置防火墙

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

**步骤二：配置客户端**

1）安装软件包

1. **[**root@client **~]**# yum –y install chrony

2) 修改配置文件

1. **[**root@client **~]**# vim **/**etc**/**chrony**.**conf
2. server **192.168.4.5** iburst                //设置与哪台服务器同步数据
3. //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

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

4) 重启chrony与服务器同步时间

1. **[**root@client **~]**# systemctl restart chronyd

5) 确认时间是否已经同步

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

## **5 案例5：pssh远程套件工具**

### **5.1 问题**

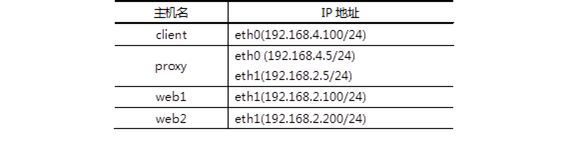
本案例要求使用pssh套件工具并发远程其他主机，具体要求如下：

* 使用密码批量、多并发远程其他主机
* 使用密钥批量、多并发远程其他主机
* 批量、多并发拷贝数据到其他主机
* 批量、多并发从其他主机下载数据到本机
* 批量、多并发杀死其他主机的进程

### **5.2 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-5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－5 主机列表



安装pssh后会提供如下命令：

/usr/bin/pnuke

/usr/bin/prsync

/usr/bin/pscp.pssh

/usr/bin/pslurp

/usr/bin/pssh

### **5.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：准备工作**

1）安装软件包

1. **[**root@proxy **~]**# rpm **-**ivh pssh**-2.3.1-5.**el7**.**noarch**.**rpm

2)修改/etc/hosts本地解析文件

1. cat **/**etc**/**hosts
2. … …
3. **192.168.2.100** host1
4. **192.168.2.200** host2
5. **192.168.4.100** host3
6. … …

3)创建主机列表文件

1. **[**root@proxy **~]**# cat **/**root**/**host**.**txt            //每行一个用户名、IP或域名
2. … …
3. root@host1
4. host2
5. host3
6. … …

**步骤二：使用密码批量、多并发远程其他主机**

1）语法格式

1. **[**root@proxy **~]**# man pssh                    //通过man帮助查看工具选项的作用
2. pssh提供并发远程连接功能
3. **-**A                使用密码远程其他主机（默认使用密钥）
4. **-**i                将输出显示在屏幕
5. **-**H                设置需要连接的主机
6. **-**h                设置主机列表文件
7. **-**p                设置并发数量
8. **-**t                设置超时时间
9. **-**o dir            设置标准输出信息保存的目录
10. **-**e dir            设置错误输出信息保存的目录
11. **-**x                传递参数给ssh

2)使用密码远程多台主机执行命令，屏幕显示标准和错误输出信息

1. **[**root@proxy **~]**# pssh **-**i **-**A **-**H 'host1 host2 host3' **\**
2. **>** **-**x '-o StrictHostKeyChecking=no' echo hello

3)使用密码远程多台主机执行命令，不显示标准和错误输出信息，通过读取host.txt读取主机信息

1. **[**root@proxy **~]**# pssh **-**A **-**h host**.**txt echo hello

**步骤三：使用密钥批量、多并发远程其他主机**

1）生成密钥并发送密钥到其他主机

1. **[**root@proxy **~]**# ssh**-**keygen **-**N '' **-**f **/**root**/.**ssh**/**id\_rsa //非交互生成密钥文件
2. **[**root@proxy **~]**# ssh**-**copy**-**id host1
3. **[**root@proxy **~]**# ssh**-**copy**-**id host2
4. **[**root@proxy **~]**# ssh**-**copy**-**id host3

2)使用密钥远程其他主机

1. **[**root@proxy **~]**# pssh **-**h host**.**txt echo hello

3)使用密钥远程其他主机，将标准输出信息写入到/tmp目录

1. **[**root@proxy **~]**# pssh **-**h host**.**txt **-**o **/**tmp**/** echo hello

**步骤四：批量、多并发拷贝数据到其他主机**

1）语法格式

1. **[**root@proxy **~]**# man pscp**.**pssh                    //通过man帮助查看工具选项的作用
2. pscp**.**pssh提供并发拷贝文件功能
3. **-**r    递归拷贝目录
4. 其他选项基本与pssh一致

2)将本地的/etc/hosts拷贝到远程主机的/tmp目录下

1. **[**root@proxy **~]**# pscp**.**pssh **-**h host**.**txt **/**etc**/**hosts **/** tmp

3)递归将本地的/etc目录拷贝到远程主机的/tmp目录下

1. **[**root@proxy **~]**# pscp**.**pssh **-**r **-**h host**.**txt **/**etc **/**tmp

**步骤五：批量、多并发从其他主机下载数据到本机**

1）语法格式

1. **[**root@proxy **~]**# man pslurp                    //通过man帮助查看工具选项的作用
2. pslurp提供远程下载功能
3. 选项与pscp**.**pssh基本一致

2)将远程主机的/etc/passwd，拷贝到当前目录下，存放在对应IP下的pass文件中

1. **[**root@proxy **~]**# pslurp **-**h host**.**txt **/**etc**/**passwd **/**pass

注意：最后的pass是文件名

3)将远程主机的/etc/passwd目录，拷贝到media下，存放在对应IP下的pass文件

1. **[**root@proxy **~]**# pslurp **-**h host**.**txt **-**L **/**media **/**etc**/**passwd **/**pass

**步骤六：批量、多并发杀死其他主机的进程**

1）语法格式

1. **[**root@proxy **~]**# man pnuke                    //通过man帮助查看工具选项的作用
2. pnuke提供远程杀死进程的功能
3. 选项与pssh基本一致

2)将远程主机上的sleep进程杀死

1. **[**root@proxy **~]**# pnuke **-**h host**.**txt sleep

3)将远程主机上的test相关脚本都杀死（如：test1,testtt,test2等等）

1. **[**root@proxy **~]**# pnuke **-**h host**.**txt test

4)将远程主机上的test.sh脚本杀死

1. **[**root@proxy **~]**# pnuke **-**h host**.**txt test**.**sh