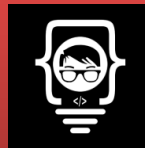




Ubuntu Server 从入门到精通

第五章：网络基础



网络

互联网就是网络的网络

- 为实现资源共享，彼此互联的多个计算设备就形成了网络
- 为实现通信，设备间必须共同遵守相同通信协议
 - TCP/IP协议族
 - 标识彼此地址，收发和处理相同约定的数据包
 - 分组交换即包交换网络
 - 分层头部、数据
 - 速度由频率决定
- 网络分层
 - 物理层、网络（internet）层、传输层、应用层
 - 将复杂的问题分解为多层的简单问题，层间遵守相同接口





网络

互联网就是网络的网络

- 网卡接口
 - enp2s0、eth0
 - ifconfig -a
 - ip link / ip address
 - sudo lshw -class network
- 管理网卡
 - sudo ethtool enp0s3
 - sudo ethtool -s duplex half|full speed 1000



网络配置

互联网就是网络的网络

- 网卡配置文件

- /etc/netwok/interfaces
 - auto eth0
 - iface eth0 inet static|DHCP
 - pre-up /sbin/ethtool -s eth0 speed 1000 duplex full

- 网络基本设置

- `sudo ifconfig eth0 10.0.0.100 netmask 255.255.255.0` 临时设置IP地址
- `sudo ifconfig eth0 10.0.0.100/24`
- `sudo route add default gw 10.0.0.1 eth0` 设置网关
- `sudo route add -net 0.0.0.0 netmask 0.0.0.0 gw 1.1.1.1` 网段路由
- `sudo route add -host 2.2.2.2 gw 1.1.1.1` 主机路由





网络配置

互联网就是网络的网络

- `/etc/resolv.conf`
- `ip addr flush eth0`
- `sudo ifconfig eth0 down`
- `sudo ifconfig eth0 up`
- `sudo systemctl restart networking.service`
- 查看路由
 - `route -n`
 - `netstat -nr`

DNS设置（软链接）

清除网卡配置

禁用网卡

启用网卡

重启网络服务



IP地址

互联网就是网络的网络

- 动态获取IP地址

- `sudo vi /etc/network/interfaces`
 - `auto eth0`
 - `iface eth0 inet dhcp`
- `sudo dhclient eth0`

查看网络配置文件

手动获取地址

- 静态IP地址

- `sudo vi /etc/network/interfaces`
 - `iface eth0 inet static`
 - `address 192.168.1.1`
 - `netmask 255.255.255.0`
 - `gateway 192.168.1.254`

查看网络配置文件



网络配置

互联网就是网络的网络

- 更多网卡配置

- network 192.168.20.0
- broadcast 192.168.20.255
- # dns-* options are implemented by the resolvconf package, if installed
- dns-nameservers 192.168.60.101 192.168.60.102
- dns-search example.com sales.example.com dev.example.com
- up route add -net 172.16.5.0/24 gw 192.168.10.100 eth1
- up route add -net 172.24.0.0/24 gw 192.168.10.100 eth1
- down route del -net 172.24.0.0/24
- down route del -net 172.16.5.0/24
- mtu 1500
- hwaddress 00:11:22:33:44:55



名称解析

互联网就是网络的网络

• 主机名解析

– `sudo vi /etc/hosts`

- 127.0.0.1 localhost
- 127.0.1.1 ubuntu-server
- 10.0.0.11 server1 server1.example.com

优先级高于resolv.conf

– `/etc/nsswitch.conf`

- files
- Resolve
- [NOTFOUND=return]
- dns
- mdns4_minimal

名称解析顺序配置文件

`/etc/hosts`

`systemd-resolved.service`(缓存、localhost、本机名)

结果即权威

DNS服务器

Multicast DNS





网桥（桥接）

把服务器当交换机用实在是大材小用

- 将多个以太网段以上层协议透明的方式连接在一起
 - 二层转发，对三层协议透明
 - 启用防火墙可对流量过滤
 - 桥接宿主机与虚拟机网络，使虚拟机访问外部网络
 - 桥接有线网与无线网
 - 链路冗余容错（需启用STP）
 - 通过网桥管理工具实现 `bridge-utils`



网桥（桥接）

把服务器当交换机用实在是大材小用

- 安装网桥管理包
 - `sudo apt install bridge-utils`
- 临时配置
 - `sudo brctl addbr br0`
 - `sudo brctl addif br0 eth0 eth1`
 - `sudo ifconfig eth0 0.0.0.0 up`
 - `sudo ifconfig eth1 0.0.0.0 up`
 - `sudo ifconfig br0 1.1.1.1/24 up`
 - `sudo dhclient br0`
 - `sudo route add default gw 1.1.1.10`



网桥（桥接）

把服务器当交换机用实在是大材小用

- 持久配置
 - vi /etc/network/interfaces
 - auto eth0
 - iface eth0 inet manual
 - auto eth1
 - iface eth1 inet manual
 - auto br0
 - iface br0 inet static
 - address 1.1.1.1
 - netmask 255.255.255.0
 - gateway 1.1.1.10
 - bridge_ports eth0 eth1
 - bridge_stp off





网桥（桥接）

把服务器当交换机用实在是大材小用

- 重启服务
 - `sudo systemctl restart networking`
 - `sudo services networking restart`
 - `sudo /etc/init.d/networking restart`
- 查看网桥信息
 - `brctl show`
 - `brctl showmacs br0`
 - `brctl showstp br0`



网卡绑定

还记得小时候这段一把筷子的故事吗

- Bonding==Port Trunking==Link aggregation==Teaming
- 将多个物理网卡组合为一个逻辑网卡
 - 高可用、负载平衡、高吞吐量
- 配置
 - `sudo echo bonding >> /etc/modules`
 - `sudo modprobe bonding`
 - `sudo systemctl stop networking`
 - `sudo vi /etc/network/interfaces`

添加内核支持
手动加载内核





网卡绑定

还记得小时候这段一把筷子的故事吗

- Mode 1 配置

- auto eth0
- iface eth0 inet manual
- bond-master bond0
- bond-primary eth0
- auto bond0
- iface bond0 inet dhcp
- bond-mode active-backup
- bond-miimon 100
- bond-slaves none

其他网卡配置项同

只有Active网卡需要

也可配置静态地址

也可使用 mode 编号

故障检测间隔

eth0配置中已经声明Primary



网卡绑定

还记得小时候这段一把筷子的故事吗

- Mode 0: round-robin
 - 网络流量（数据包）顺序平均分配给Bond中所有物理网卡
 - 高可用、负载均衡
- Mode 1: ative-backup
 - Bond中只有一个网卡Active，其他网卡全部Stanby
 - 对外只有一个网卡的MAC地址可见
 - 高可用
- mode2: balance-XOR
 - 根据源目的MAC/IP/Port进行计算，确定从哪个网卡发出（性能优于Mode 0）
 - 高可用、负载均衡



网卡绑定

还记得小时候这段一把筷子的故事吗

- Mode 3: broadcast
 - 发包广播给Bond中所有网卡，提供最短的故障恢复时间，应用连接不中断
 - 高可用
- Mode 4: 802.3ad (Dynamic link aggregation)
 - 链路聚合LACP组内的网卡使用相同速率、双工设置
 - 要求：计算机安装ethtool；交换机支持IEEE 802.3ad标准，并进行额外配置
 - 高可用、负载均衡
- mode5: balance-tlb (Adaptive transmit load balancing)
 - 隧道绑定不需要上联交换机额外配置，根据网卡负载出站负载均衡
 - 高可用、负载均衡



网卡绑定

还记得小时候这段一把筷子的故事吗

- Mode 6: balance-alb (Adaptive load balancing)
 - Mode 5 + balance-rlb (入站流量负载均衡)
 - Bond驱动拦截本机的ARP响应包，使用不同网卡硬件MAC替换源MAC
 - 不同的对端使用不同的服务器MAC地址，实现入站负载均衡
 - 不需要上连交换机额外配置
- 查看bond端口信息
 - `cat /proc/net/bonding/bond0`





网卡绑定

还记得小时候这段一把筷子的故事吗

- Mode 4 配置

- auto eth0
- iface eth0 inet manual
- bond-master bond0
- auto bond0
- iface bond0 inet dhcp
- bond-mode 4
- bond-miimon 100
- bond-lacp-rate 1
- bond-slaves eth0 eth1

其他物理网卡配置项同

每1秒发送LACPDU (默认 0, 即30秒)



DHCP服务

自动分配网络设置

- Dynamic Host Configuration Protocol
 - 透明的配置网络参数
 - IP/掩码、网关、DNS、域名、主机名、时间服务器、打印服务器
 - 通过地址租约循环使用IP地址
 - UDP 67 / 68
- 安装
 - `sudo apt install isc-dhcp-server`
 - `sudo vi /etc/default/isc-dhcp-server` # 指定启动DHCP服务的网卡
`INTERFACES="eth0"`
 - `sudo vi /etc/dhcp/dhcpd.conf` # 主配置文件（指定地址池和选项）



DHCP主配置文件

自动分配网络设置

- `default-lease-time 3600;` # 租约期限
- `max-lease-time 7200;` # 最大租约
- `authoritative;` # 授权服务器
- `subnet 192.168.1.0 netmask 255.255.255.0 {`
 - `range 192.168.1.150 192.168.1.200;`
 - `option routers 192.168.1.254;`
 - `option domain-name-servers 192.168.1.1, 192.168.1.2;`
 - `option ntp-servers 1.1.1.1;`
 - `option domain-name "local.lan";`
- `}`





DHCP地址保留

自动分配网络设置

- host name {
 - hardware ethernet 00:11:22:33:44:55;
 - fixed-address 192.168.1.11;
- }

#主机地址保留

#指定网卡MAC地址

#子网内地址，可为range外地址



DHCP主配置文件

自动分配网络设置

- 日志与状态查询

- `cat /var/lib/dhcp/dhcpd.leases`
- `tail -f /var/log/syslog`
- `systemctl status isc-dhcp-server.service`
- `less /var/lib/dhcp/dhclient.leases`

#服务器地址租约结果

#服务器日志

#服务运行状态

#客户端获得地址





NTP

时间都去哪了

- 我们每天都在用的网络时间协议
- 计时方法
 - 太阳照影、滴水、烧香、机械、电子、石英、原子时钟
 - Drift 是计时器时间与真实时间之间的偏移量
 - 基于铯133的原子时钟每3亿年误差为1秒
- 时间标准
 - GMT: 格林威治标准时间
 - UTC: 世界协调时间
 - CST: China Standard Time UT+8:00





NTP

时间都去哪了

- 跨时区沟通呼唤统一的时间定义
- 计算机技术对时间非常敏感
 - IPSec、AD、SSL.....
 - 日志审计
 - 电子元器件相互干扰加大时间偏移
- 如何保证时间准确
 - 不停地同步时间
 - 永远无法精确同步（网络通信延时影响时间同步精度）



- NTP协议的分层架构
 - 从核心向外0-16层 stratum (地壳)
 - 0 代表时间源
 - 1-15代表逐级同步的时间服务器 (越接近0时间越精确)
 - 16表示尚未同步 (不作为时间同步源)
 - 客户端服务器全部使用udp 123端口通信
- 每个移动设备都运行NTP协议
 - 硬件时钟： RTC (主板电池供电)
 - 系统时钟： Local time



NTP

时间都去哪了

- NTP客户端
 - 客户端程序从时间服务器同步时间
 - 系统启动时自动同步时间
 - 网口激活时自动同步运行
 - 手动同步时间
- 客户端命令
 - timedatectl

查看客户端时间

```
Local time: Fri 2017-10-20 14:57:26 CST
Universal time: Fri 2017-10-20 06:57:26 UTC
RTC time: Fri 2017-10-20 06:57:27
Time zone: Asia/Harbin (CST, +0800)
Network time on: yes
NTP synchronized: no
RTC in local TZ: no
```



- ntp客户端

- 新版系统使用timesyncd客户端同步时间
- `timedatectl list-timezones` #列出所有时区
- `timedatectl set-timezone` #设置时区
- `timedatectl set-time "2012-10-30 18:17:16"` #设置系统时间
- `timedatectl set-ntp true` #开启网络时间同步服务
- `systemctl status systemd-timesyncd.service` #查看时间同步服务运行状态
- `sudo hwclock -w` #将系统时间写入硬件时间
- `sudo hwclock -s` #将硬件时间写入系统时间
- `hwclock --set --date='2011-08-14 16:45:05'` #设置硬件时钟



NTP

时间都去哪了

- ntpdate
 - `sudo apt-get install ntpdate` #安装ntpdate
 - `sudo ntpdate ntp.ubuntu.com` #向指定服务器发起时间同步请求
 - `sudo ntpdate -d 1.1.1.1` #显示时间同步详细过程
 - `sudo ntpdate -q 1.1.1.1` #只查询时间，并不本地更新时间
 - `sudo ntpdate -u 1.1.1.1` #随机源端口
 - 一旦安装ntpdate / ntp，timedatectl 将被禁用
- 新版系统使用timesyncd替换ntpd的客户端功能
 - `/etc/systemd/timesyncd.conf` #配置文件



NTPD

时间都去哪了

- ntpd: 客户端 + 服务器

- apt-get install ntp
- systemctl status ntp
- systemctl restart ntp
- vi /etc/ntp.conf

- server 1.1.1.1
- fudge 127.127.1.1 stratum 10

#安装ntp服务

#查询服务状态

#重启服务

#配置文件

#使用本机时钟作为备用时间源



NTPD

时间都去哪了

- ntpq -p
 - remote # 本机正在连接的上级时间服务器
 - refid # 上级服务器的上级时间服务器
 - st # 服务器stratum层级
 - t # 协议类型：unicast, broadcast, multicast, anycast
 - when # 上一次查询服务器已过去的时间（秒）
 - poll # 查询服务器的时间间隔（ $64=2^6$ ）
 - reach # 最近8次查询结果成功则值为377（8进制数），开始时间服务
 - delay # 请求和响应之间的时间差（毫秒）
 - offset # 本地时钟与时间服务器的时间便宜
 - jitter # 与服务器的网络延时，此值应该小于100

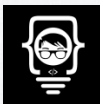




NTPD

时间都去哪了

- `ntpq -p`
- 每行第一个字符
 - 空表示无效主机
 - `x` 已不再使用
 - `-` 已不再使用
 - `#` 状态良好但未使用
 - `+` 良好且优先使用
 - `*` 首选主同步主机





NTPD

时间都去哪了

- 其他命令
 - `date --set 1998-11-02` # 设置日期
 - `date --set 21:08:0` # 设置时间
 - `cat /etc/timezone` # 查看时区



...

Questions?

