# Linux 的網路連線設定

Linux TCP/IP的網路連線設定最少需知道三項設定值分別是：IP、NETMASK、Default Gateway，網路的設定必須使用root 系統管理帳號透過兩個工具，分別是：ifconfig、route。而 Debian GNU/Linux 網路卡設定檔案也只允許root系統管理者帳號撰寫，檔案位於/etc/network/interfaces。Debian GNU/Linux 的網路卡設定檔為/etc/network/interfaces

|  |
| --- |
| ifconfig 語法為: ifconfig eth0 $IP netmask $NETMASK  如果IP = 10.1.1.30，NETMASK = 255.255.255.0  則命令為 $ ifconfig eth0 10.1.1.30 netmask 255.255.255.0  route設定default gateway語法為: route add default gw $GATEWAY  如果 GATEWAY = 10.1.1.1  則命令為: $ route add default gw 10.1.1.1  請試試看以下命令 :  $ ifconfig # 列出目前的網路卡設定狀態  $ route -n # 秀出目前的 route table 表 |

單純以手動的方式使用 ifconfig及route於系統重開機時就會還原為初始值，必須透過撰寫網路設定檔，於開機的時候初始化設定。不同公司出品的Linux都會有所不同，相同的都是透過ifconfig及route來設定網路，不過設定檔格式及位置會不同。Debian GNU/Linux網路卡設定檔位於「/etc/network/interfaces」。其格式如下。

|  |
| --- |
| $ edit /etc/network/interfaces  # /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)  # The loopback interface  # automatically added when upgrading  auto lo  iface lo inet loopback  # The first network card - this entry was created during the Debian installation  # (network, broadcast and gateway are optional)  # automatically added when upgrading  auto eth0  iface eth0 inet static  address 172.17.28.251  netmask 255.255.254.0  gateway 172.17.28.1  dns-nameservers 172.17.17.1  $ /etc/init.d/networking stop # 停掉網路裝置  $ ifconfig # 你會發現空空如也  $ /etc/init.d/networking start # 啟動網路裝置  $ ifconfig # 網路裝置回覆  $ route -n # 網路裝置回覆 |

**19.2 Client 端的設定**

**19.2.1 相關設定檔**

主機名稱對應到IP是透過DNS架構！那麼這兩種方法分別使用什麼設定檔？先來談一談幾個設定檔吧！

* /etc/hosts：這個是最早的 hostname 對應 IP 的檔案；
* /etc/resolv.conf：這個重要！就是ISP的DNS伺服器IP記錄處；
* /etc/nsswitch.conf：這個檔案則是在『決定』先要使用/etc/hosts還是/etc/resolv.conf的設定！

Linux的預設主機名稱與 IP的對應搜尋都以 /etc/hosts 為優先，查看/etc/nsswitch.conf，找到hosts的項目：

|  |
| --- |
| [root@www ~]# **vim /etc/nsswitch.conf**  hosts: files dns |

上面的files是使用/etc/hosts，而最後的dns則是使用/etc/resolv.conf的DNS伺服器來進行搜尋！因此，你可以先以 /etc/hosts 來設定IP對應！當然也可以調換過來，不過總是/etc/hosts比較簡單，所以將他擺在前面比較好！要進行DNS的測試，那要瞭解/etc/resolv.conf 的內容，假設在台灣使用hinet的168.95.1.1這部DNS伺服器，所以應該這樣寫：

|  |
| --- |
| [root@www ~]# **vim /etc/resolv.conf**  nameserver 168.95.1.1  nameserver 139.175.10.20 |

DNS伺服器的IP可設定多個，因為當第一部(照設定的順序)DNS掛點時，用戶端可以使用第二部(上述是139.175.10.20)進行查詢。通常建議至少填寫兩部DNS伺服器的IP，不過在網路正常使用的情況下，**永遠只有第一部DNS伺服器會被使用來查詢**，其他的設定值只是在第一部出問題時才會被使用。

**DNS的正、反解查詢指令： host**

|  |
| --- |
| [root@www ~]# **host [-a] FQDN [server]**  [root@www ~]# **host -l domain [server]**  選項與參數：  -a ：代表列出該主機所有的相關資訊，包括 IP、TTL 與除錯訊息等等  -l ：若後面接的那個 domain 設定允許 allow-transfer 時，則列出該 domain所管理的所有主機名稱對應資料！  server：這個參數可有可無，當想要用非/etc/resolv.conf內的DNS主機來查詢主機名稱與IP的對應時，可用這個參數！  # 1. 使用預設值來查出 linux.vbird.org 的 IP  [root@www ~]# **host linux.vbird.org**  linux.vbird.org has address 140.116.44.180 <==這是 IP  linux.vbird.org mail is handled by 10 linux.vbird.org. <==這是 MX (後續章節說明)  Received 86 bytes from 168.95.1.1#53 in 15 ms <==果然是從 168.95.1.1 取得的資料  # 2. 強制以 139.175.10.20 這部 DNS 主機來查詢  [root@www ~]# **host linux.vbird.org 139.175.10.20**  Using domain server:  Name: 139.175.10.20  Address: 139.175.10.20#53  Aliases:  linux.vbird.org has address 140.116.44.180  linux.vbird.org mail is handled by 10 linux.vbird.org. |

**DNS的正、反解查詢指令： nslookup**

|  |
| --- |
| [root@www ~]# **nslookup [FQDN] [server]**  [root@www ~]# **nslookup**  選項與參數：  1. 可以直接在 nslookup 加上待查詢的主機名稱或者是 IP ，[server] 可有可無；  2. 如果在 nslookup 後面沒有加上任何主機名稱或 IP，那將進入 nslookup的查詢功能在nslookup 的查詢功能當中，可以輸入其他參數來進行特殊查詢，例如：  set type=any ：列出所有的資訊『正解方面設定檔』  set type=mx ：列出與 mx 相關的資訊！  # 1. 直接搜尋 mail.ksu.edu.tw 的 IP 資訊  [root@www ~]# **nslookup mail.ksu.edu.tw**  Server: 168.95.1.1  Address: 168.95.1.1#53 <==還是請特別注意 DNS 的 IP 是否正確！  Non-authoritative answer:  Name: mail.ksu.edu.tw  Address: 120.114.100.20 <==回報 IP 給你囉！ |

nslookup可單純的將hostname與IP對應列出，不過還是會將查詢的DNS主機的IP列出來的！如果想要知道更多詳細的參數，那可以直接進入 nslookup 這個軟體的操作畫面中，如下範例：

# Linux 如何指定 DNS server

所有的Linux包含FreeBSD指定DNS server的方法均相同，全部都是透過撰寫「/etc/resolv.conf」，來指定DNS server 以及設定search domain。(設定 search domain 的格式 FreeBSD 與 Linux 有些微差異)

|  |
| --- |
| $ edit /etc/resolv.conf  search kh.coventive.com coventive.com # 設定 search domain  nameserver 211.20.240.115 # 設定第一台 DNS  nameserver 192.168.22.4 # 設定第二台 DNS |

<http://www.cyberciti.biz/faq/setting-up-an-network-interfaces-file/>

# /etc/network/interfaces Ubuntu Linux networking example

**Q**. Can you explain how to setup network parameters such as IP address, subnet, dhcp etc using */etc/network/interfaces* file?

**A**. /etc/network/interfaces file contains network interface configuration information for the both Ubuntu and Debian Linux. This is where you configure how your system is connected to the network.

## Defining physical interfaces such as eth0

Lines beginning with the word "auto" are used to identify the physical interfaces to be brought up when ifup is run with the -a option. (This option is used by the system boot scripts.) Physical interface names should follow the word "auto" on the same line. There can be multiple "auto" stanzas. ifup brings the named inter faces up in the order listed. For example following example setup eth0 (first network interface card) with 192.168.1.5 IP address and gateway (router) to 192.168.1.254:  
iface eth0 inet static  
address 192.168.1.5  
netmask 255.255.255.0  
gateway 192.168.1.254

## Setup interface to dhcp

To setup eth0 to dhcp, enter:  
auto eth0  
iface eth0 inet dhcp

## Examples: How to set up interfaces

Please read our previous  [How to: Ubuntu Linux convert DHCP network configuration to static IP configuration](http://www.cyberciti.biz/tips/howto-ubuntu-linux-convert-dhcp-network-configuration-to-static-ip-configuration.html) for more information.

Following is file located at /usr/share/doc/ifupdown/examples/network-interfaces, use this file as reference (don't forget interfaces man pages for more help):

######################################################################

# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

#

# A "#" character in the very first column makes the rest of the line

# be ignored. Blank lines are ignored. Lines may be indented freely.

# A "\" character at the very end of the line indicates the next line

# should be treated as a continuation of the current one.

#

# The "pre-up", "up", "down" and "post-down" options are valid for all

# interfaces, and may be specified multiple times. All other options

# may only be specified once.

#

# See the interfaces(5) manpage for information on what options are

# available.

######################################################################

# We always want the loopback interface.

#

# auto lo

# iface lo inet loopback

# An example ethernet card setup: (broadcast and gateway are optional)

#

# auto eth0

# iface eth0 inet static

# address 192.168.0.42

# network 192.168.0.0

# netmask 255.255.255.0

# broadcast 192.168.0.255

# gateway 192.168.0.1

# A more complicated ethernet setup, with a less common netmask, and a downright

# weird broadcast address: (the "up" lines are executed verbatim when the

# interface is brought up, the "down" lines when it's brought down)

#

# auto eth0

# iface eth0 inet static

# address 192.168.1.42

# network 192.168.1.0

# netmask 255.255.255.128

# broadcast 192.168.1.0

# up route add -net 192.168.1.128 netmask 255.255.255.128 gw 192.168.1.2

# up route add default gw 192.168.1.200

# down route del default gw 192.168.1.200

# down route del -net 192.168.1.128 netmask 255.255.255.128 gw 192.168.1.2

# A more complicated ethernet setup with a single ethernet card with

# two interfaces.

# Note: This happens to work since ifconfig handles it that way, not because

# ifup/down handles the ':' any differently.

# Warning: There is a known bug if you do this, since the state will not

# be properly defined if you try to 'ifdown eth0' when both interfaces

# are up. The ifconfig program will not remove eth0 but it will be

# removed from the interfaces state so you will see it up until you execute:

# 'ifdown eth0:1 ; ifup eth0; ifdown eth0'

# BTW, this is "bug" #193679 (it's not really a bug, it's more of a

# limitation)

#

# auto eth0 eth0:1

# iface eth0 inet static

# address 192.168.0.100

# network 192.168.0.0

# netmask 255.255.255.0

# broadcast 192.168.0.255

# gateway 192.168.0.1

# iface eth0:1 inet static

# address 192.168.0.200

# network 192.168.0.0

# netmask 255.255.255.0