8.1 路由



路由的觀察則可使用[route](http://linux.vbird.org/linux_server/0140networkcommand.php#route)這個指令來查閱與設定。那麼路由的形式有哪些？你又該如何確認路由是否正確呢？

8.1.1 路由表產生的類型



每部主機都有自己的路由表，須透過自己的路由表來傳遞主機的封包到下一個路由器。傳出後，該封包就要透過下一個路由器的路由表來傳送，與自己主機的路由表就沒關係！所以如果網路上的某一部路由器設定錯誤，封包的流向會發生問題。

|  |
| --- |
| [root@www ~]# route –n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0 <== 1  169.254.0.0 0.0.0.0 255.255.0.0 U 1002 0 0 eth0 <== 2  0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0 <== 3 |

Linux下的路由表是由小排列到大網域，如上面的路由表中，路由是由『192.168.1.0/24 -->169.254.0.0/16 --> 0.0.0.0/0 (預設路由)』來排列。當主機的網路封包需要傳送時，會查閱上述的三個路由規則來瞭解如何將該封包傳送出去。其實路由表主要有這幾種情況來設計的：

* 依據網路介面產生的IP而存在的路由：例如192.168.1.0/24這個路由的存在是由於鳥哥的這部主機上面擁有192.168.1.100這個IP的關係！主機上面有幾個網路介面的存在時，該網路介面就會存在一個路由。萬一主機有兩個網路介面時，例如 192.168.1.100, 192.168.2.100 時，那路由至少就會有：

|  |
| --- |
| [root@www ~]# ifconfig eth1 192.168.2.100  [root@www ~]# route -n  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.2.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  169.254.0.0 0.0.0.0 255.255.0.0 U 1002 0 0 eth0  0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0 |

* 手動或預設路由(default route)：  
  可用route手動給予額外的路由設定，例如預設路由(0.0.0.0/0)就是額外的路由。使用route最重要的是：『規劃的路由必須要是裝置(如eth0)或IP可以直接溝通(broadcast)的情況』才行。以上述的環境來看，僅有192.168.1.100及192.168.2.100，如果想要連接到192.168.5.254這個路由器時，下達：

|  |
| --- |
| [root@www ~]# route add -net 192.168.5.0 \  > netmask 255.255.255.0 gw 192.168.5.254  SIOCADDRT: No such process |

系統會回應沒法連接到該網域，因為網路介面與192.168.5.0/24沒關係！如果192.168.5.254真的是在實體網路連接上，並且與eth0連在一起，應該是這樣做：

|  |
| --- |
| [root@www ~]# route add -net 192.168.5.0 netmask 255.255.255.0 dev eth0  [root@www ~]# route -n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.5.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  192.168.2.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  169.254.0.0 0.0.0.0 255.255.0.0 U 1002 0 0 eth0  0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0 |

主機會直接用eth0裝置去嘗試連接192.168.5.254！上面路由輸出的重點是『Flags的G』！G表使用外部裝置作為Gateway！而Gateway(192.168.1.254)必在已存在的路由環境中。

事實上，在Linux的路由規則都是透過核心來達成的，所以這些路由表的規則都是在核心功能內！也就是在記憶體當中！

In [computer networking](http://en.wikipedia.org/wiki/Computer_networking), a **gateway** is a [node](http://en.wikipedia.org/wiki/Node_(networking)) (a [router](http://en.wikipedia.org/wiki/Router_(computing))) on a [TCP/IP](http://en.wikipedia.org/wiki/TCP/IP) network that serves as an access point to another network. A **default gateway** is the node on the [network](http://en.wikipedia.org/wiki/Computer_network) that the network software uses when an [IP address](http://en.wikipedia.org/wiki/IP_address) does not match any other routes in the [routing table](http://en.wikipedia.org/wiki/Routing_table).

In home computing configurations, an [ISP](http://en.wikipedia.org/wiki/Internet_service_provider) often provides a physical device which both connects local hardware to the [Internet](http://en.wikipedia.org/wiki/Internet) and serves as a gateway. Such devices include [DSL modems](http://en.wikipedia.org/wiki/DSL_modem) and [cable modems](http://en.wikipedia.org/wiki/Cable_modem).

In organizational systems a gateway is a node that routes the traffic from a [workstation](http://en.wikipedia.org/wiki/Workstation) to another [network segment](http://en.wikipedia.org/wiki/Network_segment). The default gateway commonly connects the internal networks and the outside network (Internet). A default gateway provides an entry point and an exit point in a network.

8.1.2 一個網卡綁多個IP：IP Alias的測試用途



eth0:0可在原本的eth0上模擬出一個虛擬介面，讓網路卡具有多個IP，具有多個IP的功能就被稱為IP Alias 。eth0:0的裝置可以透過[ifconfig](http://linux.vbird.org/linux_server/0140networkcommand.php" \l "ifconfig)或[ip](http://linux.vbird.org/linux_server/0140networkcommand.php" \l "ip_cmd)這兩個指令來達成！IP Alias最大的用途就是可以讓你用來『應急』！幾個常見的用途：

* 測試用：舉例來說IP分享器的設定通常使用WWW介面提供。IP分享器通常會給予一個私有IP亦即是192.168.0.1來讓使用者開啟WWW介面的瀏覽。如何連接上這部IP分享器呢？在不更動既有的網路環境下，可以直接利用：

|  |
| --- |
| [root@www ~]# ifconfig [device] [ IP ] netmask [netmask ip] [up|down]  [root@www ~]# ifconfig eth0:0 192.168.0.100 netmask 255.255.255.0 up |

來建立一個虛擬的網路介面，這樣就可以立刻連接上IP分享器了，也不會更動到你原本的網路參數設定值哩！

* 在實體網域中含有多個IP網域：像學校單位，原本的主機網路設定最好不要隨便修改，如果要讓同學們互通電腦資訊時，可讓每個同學透過IP Alias來設定同一網域的IP，如此大家就可在同一個網段內進行各項網路服務的測試了！
* 既有設備無法提供更多實體網卡時：如果這部主機需要連接多個網域，但該設備卻無法安裝更多的網卡時，只好勉為其難的使用IP Alias來提供不同網段的連線服務了！

所有的IP Alias都是由實體網卡模擬來的，所以啟動eth0:0時，eth0須先被啟動。當eth0被關閉後，所有eth0:n 的模擬網卡將同時也被關閉。

8.2 路由器架設



在區域網路裡面的主機可以透過廣播的方式來進行網路封包的傳送，但在不同網段內的主機想要互相連線時，就要路由器了。

8.2.1 什麼是路由器與IP分享器



路由器的主要功能是：『轉遞網路封包』！路由器會分析來源端封包的IP表頭，在表頭內找出要送達的目標IP後，透過路由器本身的路由表(routing table)來將封包向下一個目標(next hop)傳送。目前有兩種方法可以達成：

* 硬體功能：如Cisco, TP-Link, D-Link等公司都有生產硬體路由器，這些路由器內有嵌入式的作業系統，負責不同網域間的封包轉譯與轉遞等功能；
* 軟體功能：例如Linux這個作業系統的核心就有提供封包轉遞的能力。

本章節僅討論在乙太網路裡最簡單的路由器功能：連接兩個不同的網域。這個功能Linux個人電腦就可以達成了！

* 打開核心的封包轉遞 (IP forward) 功能

如同路由表，轉遞封包的能力也是Linux核心所提供，觀察核心功能的顯示檔案即可觀察核心是否有啟動封包轉遞：

|  |
| --- |
| [root@www ~]# cat /proc/sys/net/ipv4/ip\_forward  0 <== 0 代表沒有啟動， 1 代表啟動了 |

要讓該檔案的內容變成啟動值1最簡單的方是就是使用：『echo 1 > /proc/sys/net/ipv4/ip\_forward』。這個設定在下次重新開機後就會失效。建議直接修改系統設定檔的內容，用/etc/sysctl.conf來達成開機啟動封包轉遞的功能。

|  |
| --- |
| [root@www ~]# vim /etc/sysctl.conf  # 將底下這個設定值修改正確即可！ (本來值為 0 ，將它改為 1 即可)  net.ipv4.ip\_forward = 1  [root@www ~]# sysctl -p <==立刻讓該設定生效 |

sysctl指令是在核心工作時用來直接修改核心參數的一個指令，更多的功能可以參考man sysctl查詢。只要這個動作，Linux就具有最簡單的路由器功能了。由於Linux路由器的路由表設定方法的不同，通常路由器規劃其路由的方式有兩種：

* 靜態路由：直接以類似route這個指令來直接設定路由表到核心功能當中，設定值只要與網域環境相符即可。不過，當你的網域有變化時，路由器就得要重新設定；
* 動態路由：透過類似Quagga或zebra軟體的功能，這些軟體可以安裝在Linux路由器上，而這些軟體可以動態的偵測網域的變化，並直接修改 Linux 核心的路由表資訊，無須手動以route來修改路由表資訊！

其實IP分享器就是最簡單的NAT伺服器！NAT本身就是一個路由器，只是NAT比路由器多了一個『IP 轉換』的功能。

* 一般來說，路由器會有兩個網路介面，透過路由器本身的IP轉遞功能讓兩個網域可以互相溝通網路封包。如果兩個介面一邊是公共IP (public IP)但一邊是私有IP(private IP)呢？由於私有IP不能直接與公共IP溝通其路由資訊，此時就得要額外的『IP轉譯』功能；
* Linux的NAT伺服器可以透過修改封包的IP表頭資料之來源或目標IP，讓來自私有IP的封包可以轉成NAT伺服器的公共IP，就可連上Internet！所以當路由器兩端的網域分別是Public與Private IP時，才需要NAT的功能！

8.2.2 何時需要路由器



電腦數量小於數十部的小型企業是無須路由器的，只需利用hub/switch串接各部電腦，然後透過單一線路連接到Internet上即可。不過如果是超過數百部電腦的大型企業環境，環境通常需要考慮如下的狀況，因此才需要路由器的架設：

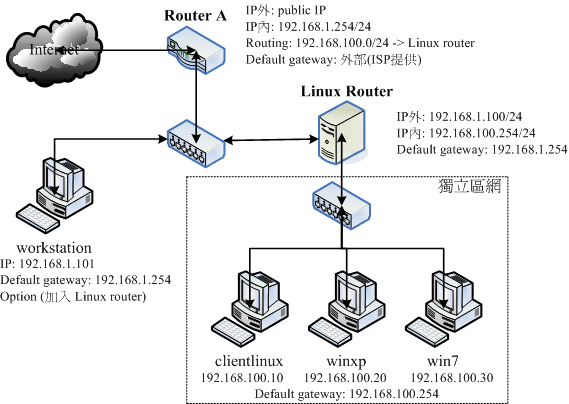
* 實體線路之佈線及效能的考量：一棟大樓的不同樓層串接所有的電腦可能有點難度，透過每個樓層架設一部路由器，並將樓層路由器相連接，就能夠簡單的管理各樓層的網路；此外，如果各樓層不想架設路由器，而直接以網路線串接各樓層的hub/switch時，由於同一網域的資料是透過廣播來傳遞的，當整個大樓的某一部電腦在廣播時，所有的電腦將會予以回應，會造成大樓內網路效能的問題；所以架設路由器將實體線路分隔，就有助於這方面的網路效能；
* 部門獨立與保護資料的考量：只要實體線路連接在一起，當資料透過廣播時，就可透過類似[tcpdump](http://linux.vbird.org/linux_server/0140networkcommand.php#tcpdump)的指令來監聽封包資料，並且予以竊取～所以如果部門之間的資料可能需要獨立，或者是某些重要的資料必須要在公司內部也予以保護時，可以將那些重要的電腦放到一獨立的實體網域，並額外加設防火牆、路由器等連接上公司內部的網域。

8.2.3 靜態路由之路由器



假設公司的網路環境中，除了一般職員的工作用電腦是直接到對外的路由器來連結網際網路，在內部還有一個部門需要安全的獨立環境，因此網路規劃可能是這樣的情況(參考[圖 3.2-1](http://linux.vbird.org/linux_server/0120intranet.php" \l "fig3.2-1)內容延伸而來)：

圖 8.2-1、靜態路由之路由器架構示意圖



以上圖的架構來說，這家公司主要有兩個class C的網段，分別是：

* 一般區網(192.168.1.0/24)：包括Router A, workstation以及Linux Router三部主機所構成；
* 保護內網(192.168.100.0/24)：包括Linux Router, clientlinux, winxp, win7等主機所構成。

192.168.1.0/24做為一般員工連接網際網路，192.168.100.0/24則給特殊部門用。workstation表一般員工的電腦，clientlinux及winxp，win7則是特殊部門的電腦，Linux Router是特殊部門連接到公司內部網域的路由器。這樣特殊部門的封包就能與公司其他部門作實體的分隔了。由上圖發現，只要是具有路由器功能的設備(Router A, Linux Router)都有兩個以上的介面，分別來溝通不同的網域，同時該路由器也都會具有一個預設路由！另外還可以加上一些防火牆的軟體在Linux Router上，以保護clientlinux, winxp, win7！如果clientlinux想連上Internet，那麼連線情況會是如何？

* 發起連線需求：clientlinux --> Linux Router --> Router A --> Internet
* 回應連線需求：Internet --> Router A --> Linux Router --> clientlinux

Router A有兩個介面，一是對外的Public IP一則是對內的Private IP，因IP類別不同，Router A需增加NAT機制。此外Router A不需要額外的設定。Linux Router將兩個網路介面卡設定兩個IP，並啟動核心的封包轉遞功能就架設完畢了！

* Linux Router

在這部主機內需要有兩張網卡，鳥哥在這裡將他定義為(假設你已經將剛剛實作的 eth0:0取消掉了)：

* eth0: 192.168.1.100/24
* eth1: 192.168.100.254/24

|  |
| --- |
| # 1. 再看看 eth0 的設定吧！雖然我們已經在第四章就搞定了：  [root@www ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth0  DEVICE="eth0"  HWADDR="08:00:27:71:85:BD"  NM\_CONTROLLED="no"  ONBOOT="yes"  BOOTPROTO=none  IPADDR=192.168.1.100  NETMASK=255.255.255.0  GATEWAY=192.168.1.254 <==最重要的設定啊！透過這部主機連出去的！  # 2. 再處理 eth1 這張之前一直都沒有驅動的網路卡吧！  [root@www ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth1  DEVICE="eth1"  HWADDR="08:00:27:2A:30:14"  NM\_CONTROLLED="no"  ONBOOT="yes"  BOOTPROTO="none"  IPADDR=192.168.100.254  NETMASK=255.255.255.0  # 3. 啟動IP轉遞，真的來實作成功才行！  [root@www ~]# vim /etc/sysctl.conf  net.ipv4.ip\_forward = 1  # 找到上述的設定值，將預設值0改為上述的1即可！儲存後離開去！  [root@www ~]# sysctl -p  [root@www ~]# cat /proc/sys/net/ipv4/ip\_forward  1 <==這就是重點！要是 1 才可以呦！  # 4. 重新啟動網路，並且觀察路由與ping Router A  [root@www ~]# /etc/init.d/network restart  [root@www ~]# route -n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.100.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0  # 上面的重點在於最後面那個路由器的設定是否正確呦！  [root@www ~]# ping -c 2 192.168.1.254  PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.  64 bytes from 192.168.1.254: icmp\_seq=1 ttl=64 time=0.294 ms  64 bytes from 192.168.1.254: icmp\_seq=2 ttl=64 time=0.119 ms <==有回應即可 |

透過ping我們也知道Linux Router可連上Router A！

* 受保護的網域，以clientlinux為例

不論clientlinux是哪一種作業系統，環境都應該是這樣的([圖 8.2-1](http://linux.vbird.org/linux_server/0230router.php#fig8.2-1))：

* IP: 192.168.100.10
* netmask: 255.255.255.0
* gateway: 192.168.100.254
* hostname: clientlinux.centos.vbird
* DNS: 168.95.1.1

|  |
| --- |
| [root@clientlinux ~]# /etc/init.d/network restart  [root@clientlinux ~]# route -n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.100.0 0.0.0.0 255.255.255.0 U 1 0 0 eth0  169.254.0.0 0.0.0.0 255.255.0.0 U 1002 0 0 eth0  0.0.0.0 192.168.100.254 0.0.0.0 UG 0 0 0 eth0  [root@clientlinux ~]# ping -c 2 192.168.100.254 <==ping自己的gateway(會成功)  [root@clientlinux ~]# ping -c 2 192.168.1.254 <==ping外部的gateway(會失敗) |

如果連ping都沒有辦法給予回應，那表示連線是有問題的！再從剛剛的回應連線需求流程來看一下！

* 發起連線：clientlinux --> Linux Router (OK) --> Router A (OK)
* 回應連線：Router A(此時router A要回應的目標是192.168.100.10)，Router A僅有public與192.168.1.0/24的路由，所以該封包會由 public介面再傳出去，因此封包就回不來了...

網路是雙向的，封包出的去回不來～告知Router A當路由規則碰到192.168.100.0/24時，要將該封包傳192.168.1.100。

* 特別的路由規則：Router A所需路由

假設Router A對外網卡為eth1，而內部的192.168.1.254設定在eth0上。怎麼在Router A增加一條路由規則呢？使用route add去加即可！如下所示的情況：

|  |
| --- |
| [root@routera ~]# route add -net 192.168.100.0 netmask 255.255.255.0 \  > gw 192.168.1.100 |

這規則不會寫到設定檔，因此下次重開機就不見了！建立一個路由設定檔依附在eth0網卡，設定檔的檔名應是route-eth0！設定檔中要設定192.168.100.0/24這網域的gateway是192.168.1.100，透過eth0，那麼寫法為：

|  |
| --- |
| [root@routera ~]# vim /etc/sysconfig/network-scripts/route-eth0  192.168.100.0/24 via 192.168.1.100 dev eth0  目標網域 透過的gateway 裝置  [root@routera ~]# route -n  Destination Gateway Genmask Flags Metric Ref Use Iface  120.114.142.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1  192.168.100.0 192.168.1.100 255.255.255.0 UG 0 0 0 eth0  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  169.254.0.0 0.0.0.0 255.255.0.0 U 0 0 0 eth1  0.0.0.0 120.114.142.254 0.0.0.0 UG 0 0 0 eth1 |

上述的重點在於192.168.100.0那行路由！如果有，ping 192.168.100.10看能不能有回應？再到clientlinux上去ping 192.168.1.254看有沒有回應，就知道設定成功！既然內部保護網路已可連上Internet，是否代表clientlinux可直接與一般員工的網域，如workstation連線呢？當clientlinux要直接連線到workstation 時，連線方向(參考[圖 8.2-1](http://linux.vbird.org/linux_server/0230router.php" \l "fig8.2-1))：

* 連線發起：clientlinux --> Linux Router (OK) --> workstation (OK)
* 回應連線：workstation (連線目標為192.168.100.10，因為並沒有該路由規則，因此丟給default gateway，亦即是 Router A) --> Router A (OK) --> Linux Router (OK) --> clientlinux

連線發起沒問題，不過回應連線會透過Router A幫忙！因為workstation 像當初的Router A，不知192.168.100.0/24在192.168.1.100裡面啦！不過Router A知道該網域在Linux Router內，所以封包還是可以順利的回到clientlinux。

* 讓workstation與clientlinux不透過Router A的溝通方式

如果不想要讓workstation透過Router A才能連線到clientlinux，就得與Router A相同，增加一條路由規則！如果是Linux 的系統，那麼如同 Router A一樣的設定如下：

|  |
| --- |
| [root@workstation ~]# vim /etc/sysconfig/network-scripts/route-eth0  192.168.100.0/24 via 192.168.1.100 dev eth0  [root@workstation ~]# /etc/init.d/network restart  [root@www ~]# route -n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  192.168.100.0 192.168.1.100 255.255.255.0 UG 0 0 0 eth0  169.254.0.0 0.0.0.0 255.255.0.0 U 0 0 0 eth0  0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0 |

最後clientlinux使用ping連到workstation，workstation也可ping到clientlinux就表示設定OK！透過這樣的設定方式可以發現一件事：『路由是雙向的，必須要瞭解出去的路由與回來時的規則』。舉例來說，在預設的情況下(Router A與workstation都沒有額外的路由設定時)，封包可由clientlinux連線到workstation的，但workstation卻沒有相關的路由可以回應到clientlinux～所以上頭才會要在Router A或者是workstation上設定額外的路由規則！

不過要提醒的是，如果Linux Router有設定防火牆，而且有設定類似NAT主機的IP偽裝技術，可能會造成路由誤判的問題～上述的Linux Router當中『並沒有使用到任何NAT的功能』！