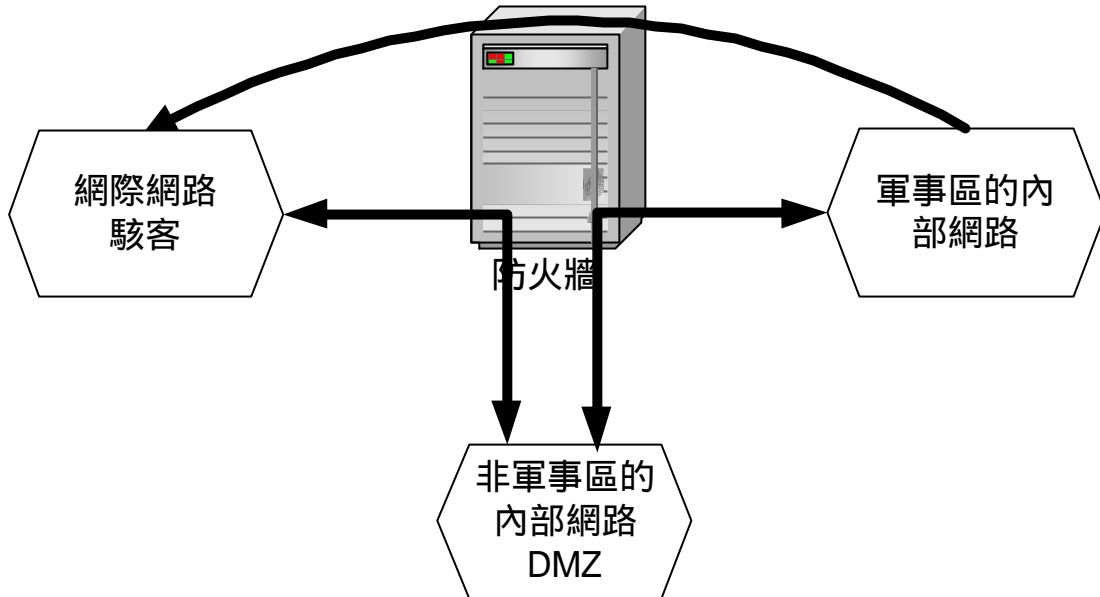




防火牆及 NAT 的設定

內部電腦對網際網路的存取



由於越多人使用網路，也因此網路成了一個社會。有很多人使用網路來作好事，但是也有少數的人使用網路犯罪。為了防止我們的網站或網域遭受到駭客的入侵破壞，我們使用防火牆將我們的網路資源分成軍事區和非軍事區(demilitarized zone 簡稱 DMZ)。軍事區的內部網路是放置受保護的網路資源，且嚴格禁止外部網際網路或駭客進入，例如資料庫、郵件伺服器....。而非軍事區放置一般可以讓外部網域使用的網路資源來供外部使用，如網站 www、FTP 檔案傳輸伺服器。我們軍事區的內部網路可以透過防火牆存取外部網際網路的資源。軍事區的內部網路和非軍事區的內部網路也可以透過防火牆的控制來互相存取資源。網際網路和非軍事區的內部網路也可以透過防火牆的控制來存取防火牆允許的資源。但是，如上圖，防火牆只允許軍事區的內部網路存取網際網路的資源，而不允許網際網路來存取軍事區內部網路的資源。

1-1 在核心設定防火牆

在安裝防火牆前，我們需要讓我們作業系統的核心 Kernel 來支援封包過濾(防火牆)。一般的 FreeBSD 核心 kernel 並不支援封包過濾(防火牆)。我們也希望在核心增加防火牆的兩個選項後再重新編譯。分別是 IPFIREWALL 選項和 IPDEVERT 選項。

首先我們進入 /usr/src/sys/i386/conf 的目錄。

```
#cd /usr/src/sys/i386/conf
```

我們將一般 GENERIC 核心複製到新的核心 newkernel。

```
#cp GENERIC newkernel
```

編輯新的核心，並且加入支援防火牆的選項 IPFIREWALL 和 IPDIVERT，IPFIREWALL_DEFAULT_TO_ACCEPT 的選項是讓核心能夠預設是接受防火牆允許的政策。

```
#vi newkernel
```

```
options IPFIREWALL
```

```
options IPDIVERT
```

```
options IPFIREWALL_DEFAULT_TO_ACCEPT
```

這是 newkernel，也就是支援防火牆的核心。

```
flash# ls
GENERIC          NOTES           SMP             newkernel
GENERIC.hints   OLDCARD        gethints.awk
Makefile         PAE            kk
```

編輯好新核心後，我們進入 /usr/src 目錄，這裏有編譯核心的 Makefile 檔

```
#cd /usr/src
```

```
[ju@flash/usr/src]# ls
COPYRIGHT      UPDATING      games/         libexec/       sys/
MAINTAINERS    bin/          gnu/          release/       tools/
Makefile       contrib/      include/      sbin/         usr.bin/
Makefile.incl  crypto/      kerberos5/   secure/        usr.sbin/
README         etc/         lib/          share/
```

我們使用 make 來編譯新的核心 newkernel。編譯核心需要花一些時間。

```
# make buildkernel KERNCONF=newkernel
```

編譯好新的核心後，我們再安裝它。

```
#make installkernel KERNCONF=newkernel
```

如果我們只是暫時性用到防火牆，而不想再重新編譯支援防火牆的核心，我們可以使用 kldload ipfw 來載入 ipfw 防火牆模組。

```
#kldload ipfw
```

1-2ipfw 防火牆指令

當我們重新組態我們的核心來提供防火牆的選項時 IPFWALL_DEFAULT_TO_ACCEPT 選項讓所有的封包都能夠流進流出。我們可以將防火牆設定為預設是允許的政策。

```
#ipfw add 65534 allow all from any to any
```

這個選項就是防火牆預設是允許的政策，它在安裝核心時，我們有選取。

```
options IPFWALL_DEFAULT_TO_ACCEPT
```

ipfw 是防火牆的管理工具，我們可以使用 `man ipfw` 來觀看 ipfw 工具的使用方法

ipfw 語法：

ipfw [-N] 命令 編號 行為 log protocol address

options

-N：一般 ipfw 使用編號報告主機和連接埠。使用這個參數引起它使用 DNS 和 /etc/service 來轉換這些編號成名稱。

命令 command：這些命令包括 add、delete、list、flush 和 resetlog。

編號 index：編號索引從 0 到 65535，假如我們省略這個參數，系統將自動產生高於這個系統 100 號的編號 index。

行為 action：這個參數指出封包應該作的行為。Allow 的行為指出封包應該被接受，Allow 是和 accept 和 pass 一樣的。deny 和 reject 會阻檔封包。deny 引起 FreeBSD 忽略封包，所以對於傳送者這個封包會遺失。Reject 引起 FreeBSD 去傳出錯誤的封包回傳送者。使用 deny 可以減輕 DoS 的攻擊，而且可以讓駭客較不容易入侵系統。

log：這個參數會產生輸出的資訊記錄檔。

protocol：這個參數指定封包的協定，包含 tcp、udp 或 icmp。ip 或 all 符合任何形態的封包。

address：address 位置包含來源和目的地 IP 位置和連接埠，就像是這網路介面。

Options：我們可以提供幾個符合封包的特別形態。established 符合存在連接的封包，setup 符合嘗試建立連接的封包。

ipfw 的命令 command :

當我們設定防火牆的規則時，我們會用到 ipfw 的命令 command。

命令	說明
add	這個 add 指令增加規則到防火牆。
delete	這個 delete 指令從防火牆刪除規則。
list	這個 list 指令從防火牆列出所有的規則。
flush	這個 flush 指令從防火牆刪除所有的規則。
resetlog	這個 resetlog 指令重設符合的計數到防火牆的規則。我們可以建立 cron 工作來週期性的重設我們想要記錄的規則，所以假如到達我們所設定的最大封包量，我們可以繼續觀看符合者。

ipfw 的 address :

這是 ipfw 的 address 參數

from address/mask [port] to address/mask [port] [via interface]

每一個 TCP/IP 的封包都包含來源和目的地的位置。我們使用 from 和 to 關鍵字來在防火牆上指定它們。我們可以使用 any 關鍵字來代替任何的位置。

來源和目的地位置都可以用單一 IP 位置來表示(61.218.29.1)，我們也可以使用網路位址加上遮罩 (61.218.29.2/24)，我們也可以使用 4 個位元組的遮罩 (61.218.29.2/255.255.255.0)。any 是指任何的網路位址。

Port 是指連接埠的編號。我們可以指定連接埠的範圍例如使用 1-1024。

假如我們希望應用在特別的網路界面，我們可以使用 via 這個關鍵字。interface 可以使用電腦的 IP 位址，也可以使用介面名稱 rl0。

1-2-1ipfw 防火牆實作範例

這個命令增加 add 一個規則 65534，這是在 65535 規則之前。這命令告訴系統接受所有來自任何地方的流量經過它再到任何的地方去，也就是不限制流量的進出。

#ipfw add 65534 allow all from any to any

當我們新增防火牆的命令與規則後，我們可以使用 ipfw list 來觀看防火牆的規則

```
[ju@flash/]# ipfw list
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
65000 allow ip from any to any
65535 allow ip from any to any
[ju@flash/]# ipfw add 65534 allow all from any to any
65534 allow ip from any to any
[ju@flash/]# ipfw list
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
65000 allow ip from any to any
65534 allow ip from any to any
65535 allow ip from any to any
```

使用 ipfw list 就可以觀看防火牆的規則。

#ipfw list

我們在編譯核心時，預設防火牆 65535 是允許任何的流量進出，我們可以新增防火牆規則來覆寫預設的規則，我們也可以將規則給刪除，我們使用 delete 指令。我們使用 ipfw delete 65534 將防火牆 65534 的規則刪除。我們使用 ipfw flush 來刪除所有新增的規則。

#ipfw delete 65534

#ipfw flush

```
[ju@flash/]# ipfw delete 65534
[ju@flash/]# ipfw flush
Are you sure? [yn] y
```

Flushed all rules.

這是已經更新後的防火牆規則，只剩下核心預設的 65535 規則。

```
[ju@flash/]# ipfw list
65535 allow ip from any to any
```

連接埠 25 是我們 SMTP 郵件伺服器 61.218.29.2 位置的連接埠，我們允許任何位置的郵件郵寄到我們的郵件伺服器上。any to 61.218.29.2 25。

#ipfw add allow tcp from any to 61.218.29.2 25

我們使用 telnet flash.aasir.com 25，就可以從遠端進入到郵件伺服器觀看，這表示遠端的郵件可以進入到我們的郵件伺服器。

```
[root@aasir chaiyen]# telnet flash.aasir.com 25
Trying 61.218.29.3...
Connected to flash.aasir.com (61.218.29.3).
Escape character is '^]'.
220 flash.aasir.com ESMTP Sendmail 8.12.9/8.12.9; Fri, 15 Aug 2003 10:14:31 +0800 (CST)
```

這是允許我們的郵件伺服器郵寄郵件到任何的地方 from 61.218.29.2 to any。

#ipfw add allow tcp from 61.218.29.2 25 to any

假如我們希望支援任何的協定，我們可以使用下列規則來允許向外連接 HTTP 和 Telnet。這個規則明確的允許只有沒有授權的連接埠才能連接到外面任何電腦的 Telnet(23)或 HTTP(80)連接埠。

#ipfw add allow tcp from 61.218.29.6 1025-65535 to any 23,80

這是允許建立連接到本地端沒有設限的連接埠。因為使用者建立連接，所有符合 established 選項的封包都會回傳，但是任何嘗試去初始化建立連接任何在高編號連接埠的伺服器將會失敗。我們希望明確的阻斷任何想存取非授權連接埠的伺服器。當我們執行 X 視窗程式，其連接埠在 6000 到 6023 之間，因此我們早一點建立規則來保護這些連接埠。我們因此使用下列防火牆的規則。

#ipfw add allow tcp from any 23,80 to 61.218.29.2 1025-65535 established

在/etc/rc.firewall 中，包含許多防火牆的規則。我們可以在那裏看到許多防火牆的實用範例。

1-2-2 防火牆的自動設定

我們在/etc/rc.conf 新增加防火牆，這樣在開機時就可以啟動。

```
#vi /etc/rc.conf
```

```
firewall_enable="YES"
```

```
firewall_type="client"
```

這些選項告訴系統在開機時執行/etc/rc.firewall 的程式，使用防火牆的型態是 client。可能的形態有 open(允許存取任何的系統)、closed(取消任何的存取)、client(將防火牆以何理的方式組態)、simple(適合 router 防火牆)和 unkonwn(只有 deny 規則)我們也可以指定包含防火牆規則的檔案名稱，這是告訴系統從這檔案載入防火牆規則(firewall_type="/usr/local/etc/firewall")。這個檔案包含 ipfw 防火牆命另減去 ipfw 指令。這看起來就像是

```
add allow udp from 61.218.29.1 to 61.218.29.6
```

```
add allow udp from 61.218.29.6 to 61.218.29.8
```

這/etc/rc.firewall 程式載入防火牆的模組，和我們所指定的防火牆規則。

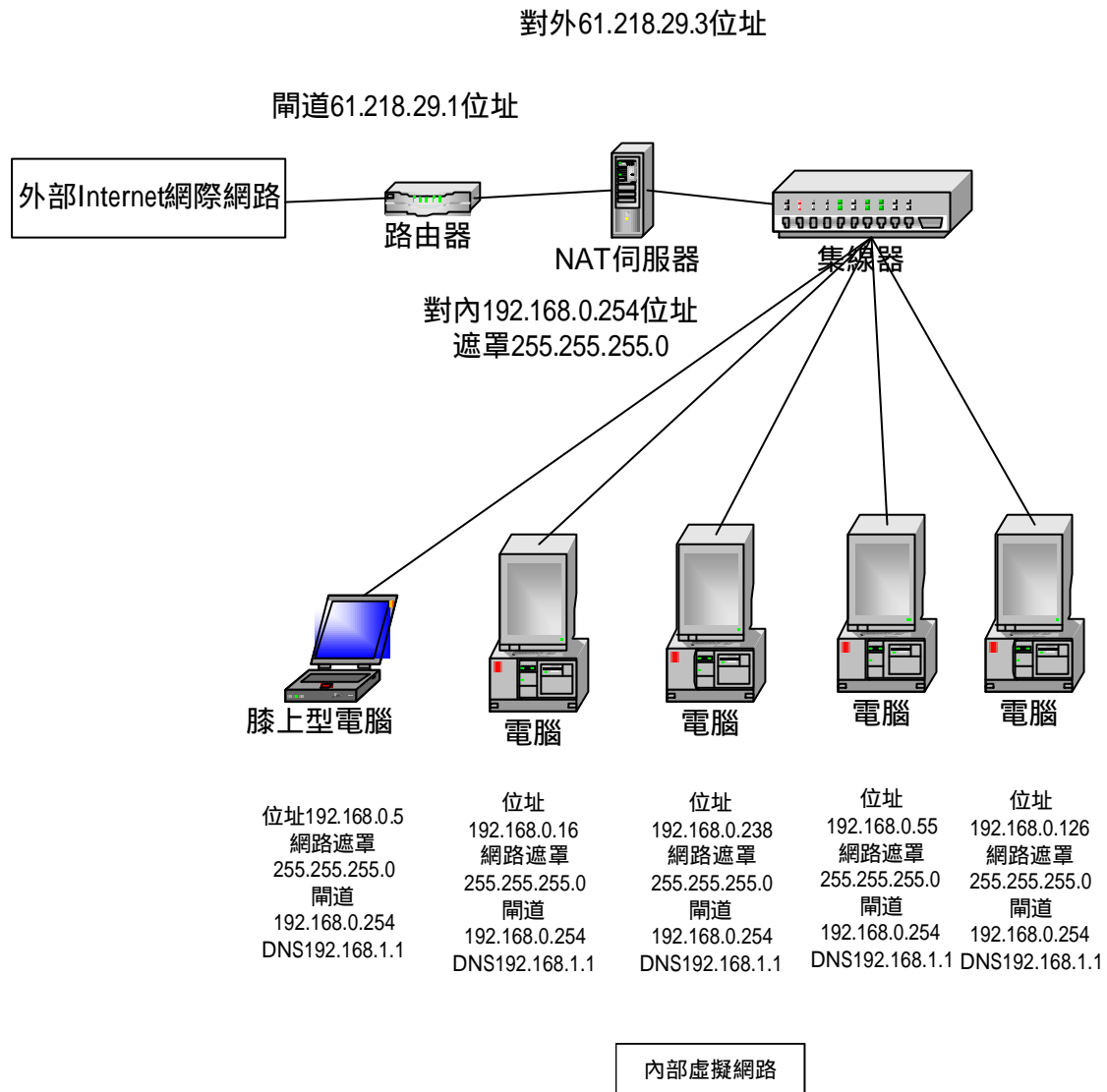
我們也可以使用/etc/netstart 來重新啟動網路組態，這包含了防火牆。假如我們的網路使用 DHCP 來分配 IP 位址，我們可以使用 killall dhclient，再使用/etc/netstart 來重新啟動。

```
[ju@flash/]# /etc/netstart
hw.bus.devctl_disable: 1 -> 1
rl0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    inet6 fe80::290:ccff:fe0d:a05e%rl0 prefixlen 64 scopeid 0x1
    inet 61.218.29.3 netmask 0xffffffff broadcast 61.218.29.7
    ether 00:90:cc:0d:a0:5e
    media: Ethernet autoselect (10baseT/UTP)
    status: active
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
    inet 127.0.0.1 netmask 0xff000000
Starting ppp as "root"
Flushed all rules.
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
ipfw: unrecognised option [-l] ip

65000 allow ip from any to any
Firewall rules loaded, starting divert daemons:.
net.inet.ip.fw.enable: 1 -> 1
add net default: gateway 61.218.29.1
Additional routing options:.
```

1-3NAT 實作

我們可以使用 NAT 伺服器當作網址偽裝(MASQUERADE)，我們可以使用一個固定 IP 位址對外，然後有多台的電腦使用虛擬 IP 位址。這樣我們就可以讓我們的多台電腦只使用一個固定 IP 或浮動 IP 就可以上網了。我們在這裏使用 flash.aasir.com 的網站來作實務應用。網站的開道為 61.218.29.1。而我們的網站 IP 為 61.218.29.3，它也是當作 NAT 伺服器，NAT 伺服器為網路轉換位址伺服器。我們對內的 IP 位址為 192.168.0.254，它的遮罩是 255.255.255.0，因此總共可有 253 個虛擬 IP，也就是 253 台電腦可以使用 61.218.29.3 個位址。在這裏我們將以 192.168.0.5 這台網址偽裝電腦來當作內部電腦的實務運用。而 61.218.29.3 的電腦當作防火牆及 NAT。



1- 3-1NAT 的組態設定

我們設定 `firewall_enable` 為 `yes` , `firewall_type` 為 `open` , 這些都是在 `/etc/rc.conf` 中作設定。

假如我們要我們 FreeBSD 系統提供 NAT 路由的功能, 我們一定要使用 `natd` 程式來作這個工作。我們可以在 `/etc/rc.conf` 中設定 NAT 的組態。

一般 NAT 路由有兩個或更多的界面卡, 一個是連接本地端的網路, 一個是連接到外面的網路 `gateway_enable` 選項啟動這兩個界面的路由(也就是讓它們相通)。`natd_interface` 選項指定網路介面連接到網路。

假如我們要執行在 NAT 後方建立伺服器, 我們就要使用 `natd_flags` 的選項。我們可以使用 `natd_flags="redirect_port tcp 192.168.0.2:80 80"` 來將 61.218.29.2 80 的位置轉向到 192.168.0.2 連接埠 80 的位置, 這樣就可以防止被攻擊。

我們編輯 `nat` 及防火牆的主機, 其有兩張網路卡, 第一張網路卡是 61.218.29.3 是對外, 第二張網路卡是 192.168.0.254 是對內。我們編輯 `/etc/rc.conf` 的組態檔。

這是設定 `r10` 第一張對外的界面卡 `Flash.aasir.com` 為我們主機的名子, `ifconfig_r10` 為第一張網路卡的 IP, 61.218.29.3, 而其網路遮罩是 255.255.255.248。

```
ifconfig_r10="inet 61.218.29.3 netmask 255.255.255.248"
```

```
defaultrouter="61.218.29.1"
```

```
hostname="flash.aasir.com"
```

這是設定第二張對內的界面卡。

```
ifconfig_r11="inet 192.168.0.254 netmask 255.255.255.0"
```

```
#vi /etc/rc.conf
```

```
gateway_enable="YES"
```

```
firewall_enable="YES"
```

```
firewall_type="open"
```

```
natd_enable="YES"
```

```
natd_interface="r10"
```

```
natd_flags=""
```

```
ifconfig_r10="inet 61.218.29.3 netmask 255.255.255.248"
```

```
defaultrouter="61.218.29.1"
```

```
hostname="flash.aasir.com"
```

```
ifconfig_r11="inet 192.168.0.254 netmask 255.255.255.0"
```

1-3-2 設定核心支援防火牆及 NAT

NAT 的功能要有作業系統核心的支援，而這不是標準作業系統所內建，因此我們要在核心設定下列兩個選項來支援 NAT，再重新編譯。前面已經有核心內件防火牆並且重新編譯。IPFIREWALL 是支援防火牆的選項，IPDIVERT 是支援 NAT 的位址轉向。

```
options IPFIREWALL
```

```
options IPDIVERT
```

1-3-3 設定防火牆組態

這是新增 NAT 防火牆的功能，這是經過 rl0 界面卡，我們預設是通過所有流進與流出的流量。ipfw -f flush 是強制刪除先前防火牆的設定。add divert natd 是增加 NAT 位置的轉向。Add pass 是讓所有進出的流量通過。

```
#vi /etc/rc.firewall
```

```
#!/bin/sh
```

```
ipfw -f flush
```

```
ipfw add divert natd all from any to any via rl0
```

```
ipfw add pass all from any to any █
```

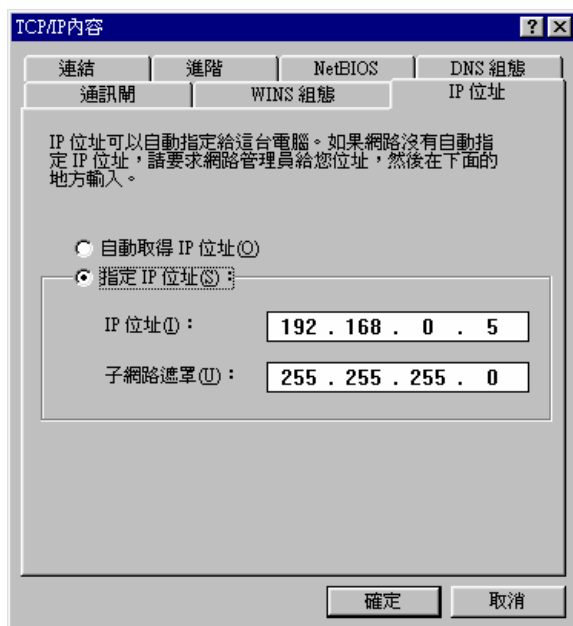
這是重新執行防火牆。

```
#sh /etc/rc.firewall
```

1-3-4 設定內部本地端虛擬網路

這是在 windows98 作設定。這是我們內部虛擬網路的電腦，它的 IP 位址為

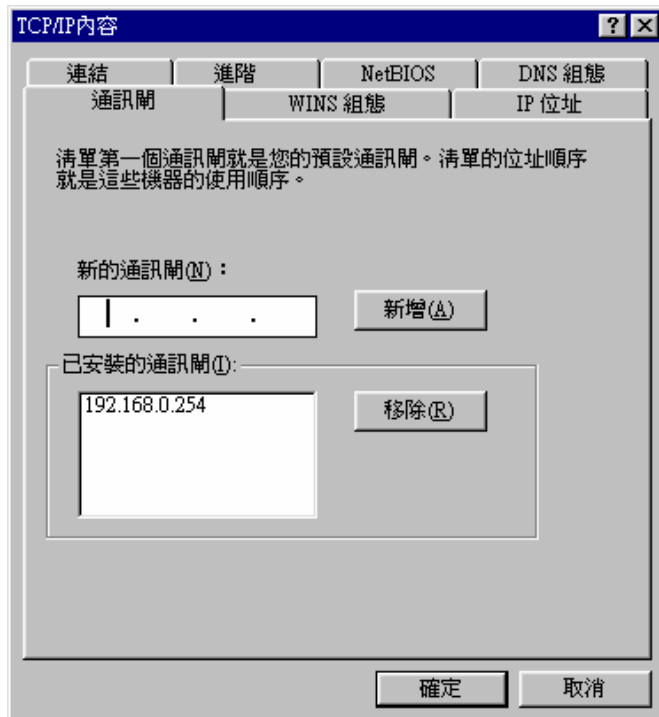
192.168.0.5。它的網路遮罩是 255.255.255.0。



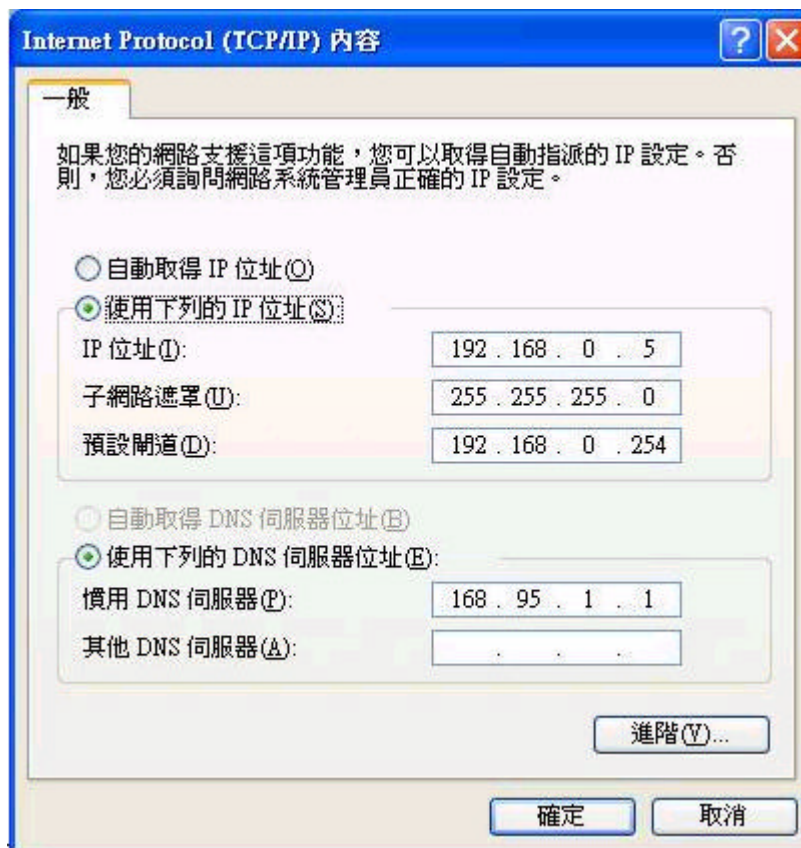
這是我們 DNS 的組態。我們要設定我們的名稱伺服器為 168.95.1.1。



我們要設定我們的通訊閘，閘道的 IP 為 192.168.0.254，也就是 NAT 網路位址轉換伺服器的內部 IP 192.168.0.254。



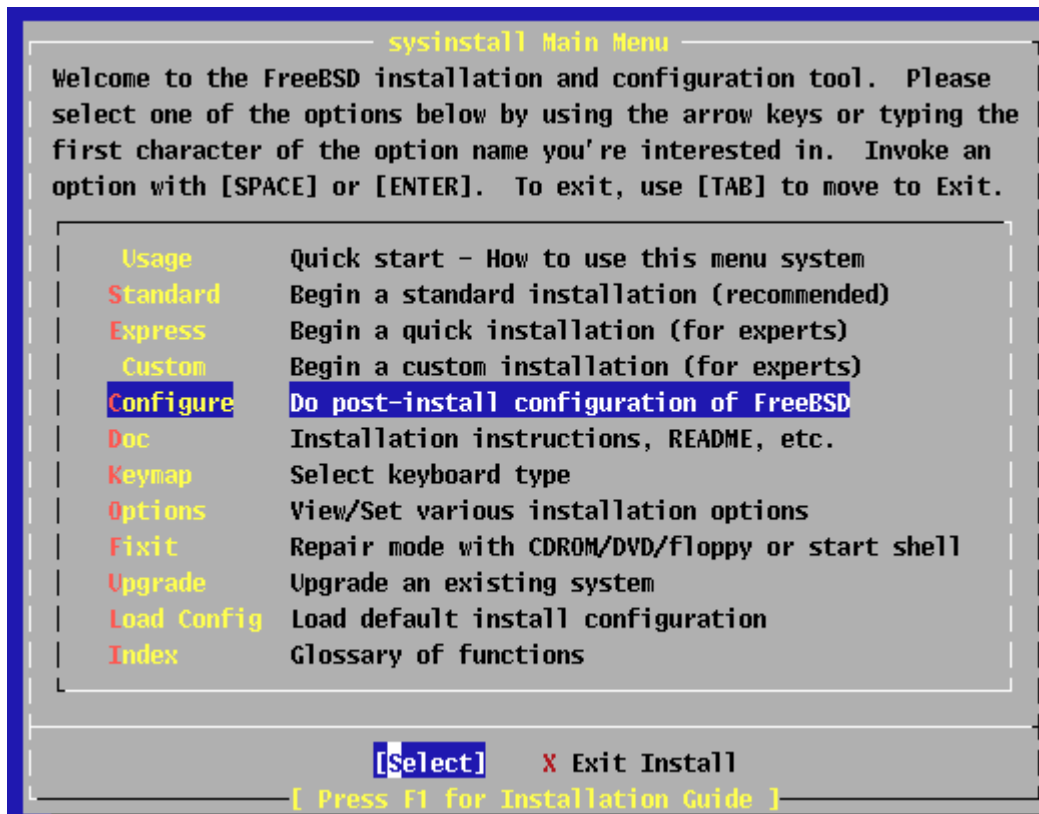
這是在 Windows XP 作設定。



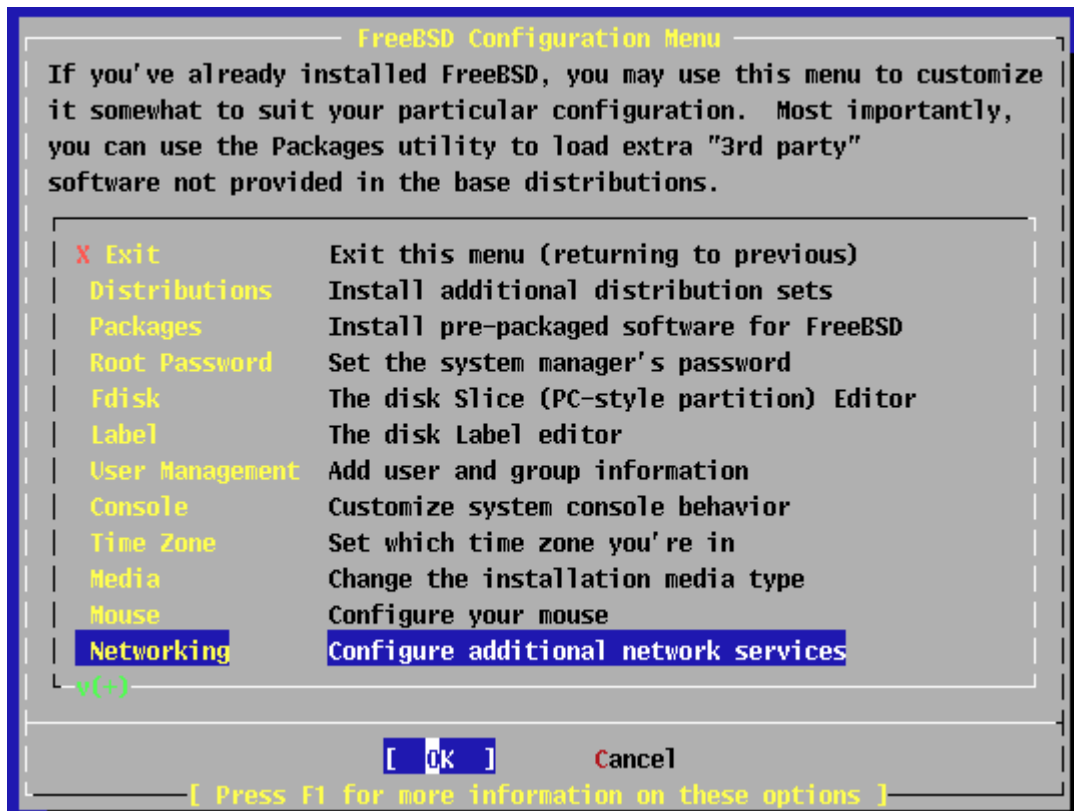
1-3-5 設定 NAT 伺服器的兩張網卡
啟動系統選項設定。

#sysinstall

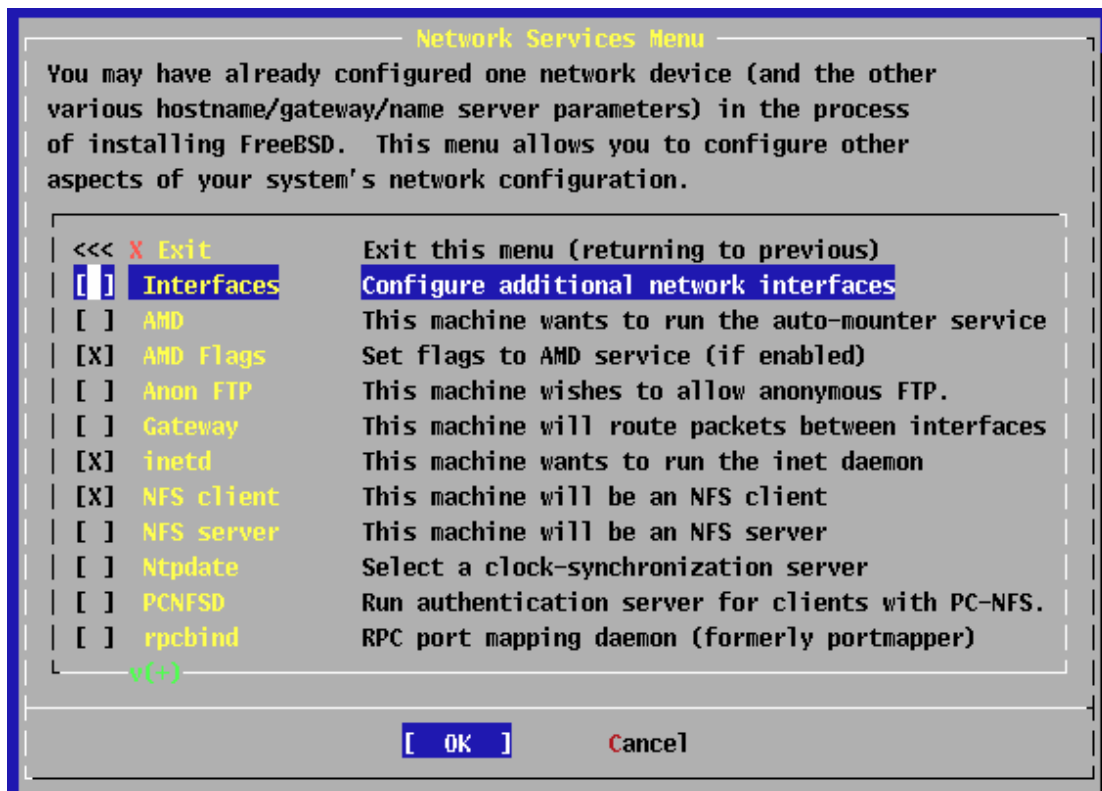
我們選取 Configure 組態。



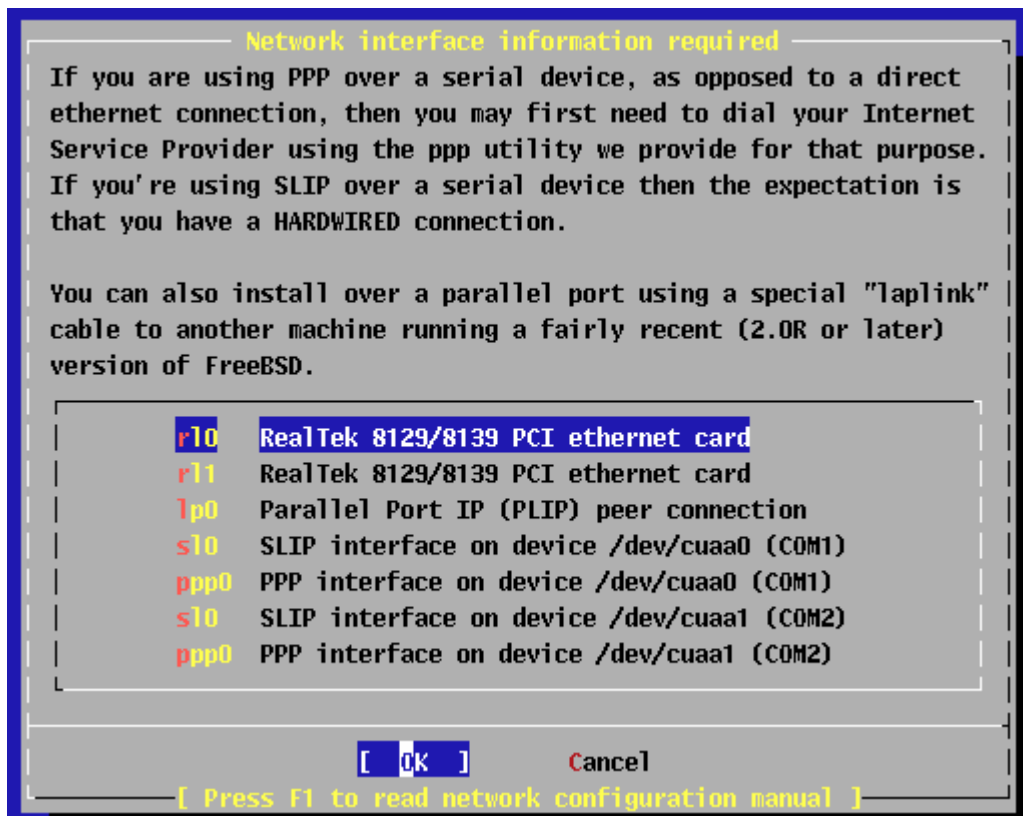
我們選取網路 Networking。



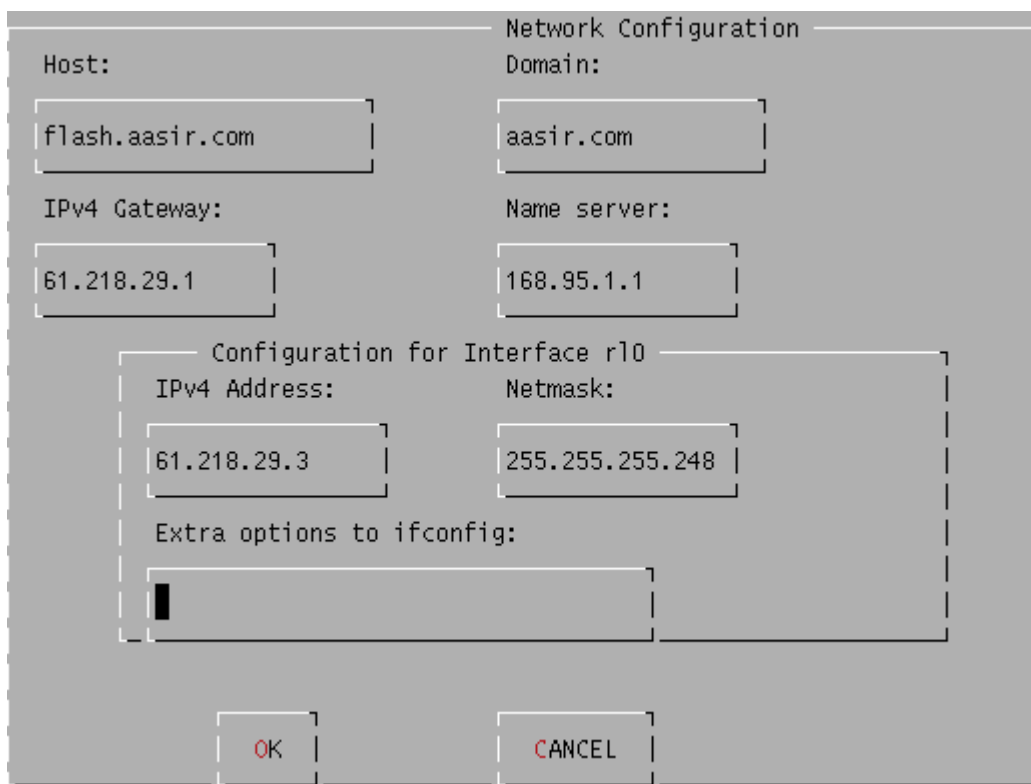
我們選取界面卡 interfaces。



在這裏有兩張網路卡 rl0 和 rl1，這表示這台機器上有兩個網路界面卡。



我們使用 r10 網卡來對外，而使用 r11 網卡來對內。r10 是對外的網路卡，它的位置是 61.218.29.3、網路遮罩是 255.255.255.248、閘道是 61.218.29.1。



第一張網路卡 r10 設定後，會自動在/etc/rc.conf 中增加下列 r10 網路卡的組態。

這是設定 r10 第一張對外的界面卡。flash.aasir.com 為我們主機的名子, ifconfig_r10 為第一張網路卡的 IP, 61.218.29.3, 而其網路遮罩是 255.255.255.248 defaultrouter 為預設閘道的位置。

#vi /etc/rc.conf

```
ifconfig_r10="inet 61.218.29.3 netmask 255.255.255.248"
```

```
defaultrouter="61.218.29.1"
```

```
hostname="flash.aasir.com"
```

我們第二張網路卡開始就從/etc/rc.conf 來設定。這是第二張網路卡 r11 的設定, 它的匿名為 r11, 而它的位址為 192.168.0.254, 而它的網路遮罩是 255.255.255.0。

#vi /etc/rc.conf

```
ifconfig_r11="inet 192.168.0.254 netmask 255.255.255.0"
```

這是兩張網路卡 r10 和 r11 裝在 NAT 伺服器上。這是 r11 和 r10 網路卡的硬體, 都是螃蟹卡 RTL8139。這是 NAT 伺服器的主機, 名稱為 flash.aasir.com, 而其 IP 為 61.218.29.3 這是 r10 網路卡 DNS 的設定, 我們設定其名稱伺服器為 168.95.1.1