**iptables 設定入門**

            一般來說，如果您的防火牆是設在 Linux 本機上面，並且您的 Linux 本機並沒有啟用 NAT 的功能，那麼您只需要考慮 filter 這個 table 的 INPUT 與 OUTPUT 這兩條鏈就可以了！簡化的很多喔！但是如果您的 Linux 主機還有考慮到 NAT 的功能，那麼 nat table 的 PREROUTING 與 POSTROUTING 還有 filter table 的 FORWARD 就得需要好好的設定一番了！

**ipchains 和 iptables 在語法上的主要的差異，注意如下：**

|  |
| --- |
| 1. 在 ipchains 中，諸如 input 鏈，是使用小寫的 chains 名，在 iptables 中，要改用大寫 INPUT。  2. 在 iptables 中，要指定規則是欲作用在那一個規則表上(使用 -t 來指定，如 -t nat)，若不指定，則預設是作用在 filter 這個表。  3. 在 ipchains 中， -i 是指介面(interface)，但在 iptables 中，-i 則是指進入的方向，且多了 -o，代表出去的方向。  4. 在 iptables 中，來源 port 要使用關鍵字 --sport 或 --source-port  5. 在 iptables 中，目的 port 要使用關鍵字 --dport 或 --destination-port  6. 在 iptables 中，"丟棄" 的處置動作，不再使用 DENY 這個 target，改用 DROP。  7. 在 ipchains 的記錄檔功能 -l，已改為目標 -j LOG，並可指定記錄檔的標題。  8. 在 ipchains 中的旗標 -y，在 iptables 中可用 --syn 或 --tcp-flag SYN,ACK,FIN SYN  9. 在 iptables 中，imcp messages 型態，要加上關鍵字 --icmp-type，如：  iptables -A OUTPUT -o eth0 -p icmp -s $FW\_IP --icmp-type 8 -d any/0 -j ACCEPT |

**觀察目前的設定**

[root@test root]# **iptables -L -n**

Chain INPUT (policy ACCEPT)   
target     prot opt source               destination   
　   
Chain FORWARD (policy ACCEPT)   
target     prot opt source               destination   
　   
Chain OUTPUT (policy ACCEPT)   
target     prot opt source               destination   
　   
# 仔細看到上面，因為沒有加上 -t 的參數，所以預設就是 filter 這個表格，   
# 在這個表格當中有三條鏈，分別是 INPUT, OUTPUT 與 FORWARD ，而且因為   
# 沒有規則，所以規則裡面都是空的！同時注意一下，在每個 chain 的後面 ()    
# 裡面，會發現有 policy 對吧！那就是『預設動作(政策)』咯！以上面來看，   
# 雖然我們啟動了 iptables ，但是我們沒有設定規則，然後政策又是 ACCEPT，   
# 所以『任何封包都會接受』的意思喔！

[root@test root]# **iptables -t nat -L -n**  
Chain PREROUTING (policy ACCEPT)   
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)   
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)   
target     prot opt source               destination   
　   
# 與 filter 類似的， nat 這個表格裡面有的則是 PREROUTING, POSTROUTING   
# 以及 OUTPUT 三條鏈！

**清除所有的規則**

一開始要先清除所有的規則，重新開始，以免舊有的規則影響新的設定。作法如下：

|  |
| --- |
| ###-----------------------------------------------------###  # 清除先前的設定  ###-----------------------------------------------------###  # 清除預設表 filter 中，所有規則鏈中的規則  iptables -F  # 清除預設表 filter 中，使用者自訂鏈中的規則  iptables -X  # 清除mangle表中，所有規則鏈中的規則  iptables -F -t mangle  # 清除mangle表中，使用者自訂鏈中的規則  iptables -t mangle -X  # 清除nat表中，所有規則鏈中的規則  iptables -F -t nat  # 清除nat表中，使用者自訂鏈中的規則  iptables -t nat -X |

**[root@test root]# /sbin/iptables -F   
[root@test root]# /sbin/iptables -X   
[root@test root]# /sbin/iptables -Z   
[root@test root]# /sbin/iptables -t nat -F**  
# 請注意，如果在遠端連線的時候，『這三個指令必須要用 scripts 來連續執行』，   
# 不然肯定『會讓您自己被主機擋在門外！』

參數說明：   
-F ：清除所有的已訂定的規則；   
-X ：殺掉所有使用者建立的 tables ；   
-Z ：將所有的 chain 的計數與流量統計都歸零

**※****選定預設的政策**

接著，要選定各個不同的規則鏈，預設的政策為何。

清除規則之後，再接下來就是要設定規則的政策啦！『當您的封包不在您設定的規則之內時，則該封包的通過與否，以 Policy 的設定為準』，通常這個政策在 filter 這個 table 的 INPUT 鏈方面可以定義的比較嚴格一點，而 FORWARD 與 OUTPUT 則可以訂定的鬆一些！通常都是將 INPUT 的 policy 定義為 DROP 啦！全部都擋掉再說！

[root@test root]# **/sbin/iptables** [-t tables] [-P] [INPUT,OUTPUT,FORWARD| PREROUTING,OUTPUT,POSTROUTING] [ACCEPT,DROP]   
參數說明：   
-P 　　：定義政策( Policy )。**注意，這個 P 為大寫啊！**   
INPUT　：封包為輸入主機的方向；   
OUTPUT ：封包為輸出主機的方向；   
FORWARD：封包為不進入主機而向外再傳輸出去的方向；   
PREROUTING ：在進入路由之前進行的工作；   
OUTPUT　 　：封包為輸出主機的方向；   
POSTROUTING：在進入路由之後進行的工作。   
範例：   
[root@test root]# **/sbin/iptables -P   INPUT DROP**  
[root@test root]# **/sbin/iptables -P  OUTPUT ACCEPT**  
[root@test root]# **/sbin/iptables -P FORWARD ACCEPT**  
[root@test root]# **/sbin/iptables -t nat -P  PREROUTING ACCEPT**  
[root@test root]# **/sbin/iptables -t nat -P      OUTPUT ACCEPT**  
[root@test root]# **/sbin/iptables -t nat -P POSTROUTING ACCEPT**  
# 除了 INPUT 之外，其他都給他設定為接受囉！在上面的設定之後，   
# 我們的主機發出的封包可以出去，但是任何封包都無法進入，   
# 包括回應給我們送出封包的回應封包(ACK)也無法進入喔！ ^\_^

**增加與插入規則**  
　   
好了，接下來準備要來定義規則了！請注意，在這個小節裡面我們完全都針對 Linux 本機來進行設定 ( 就是僅針對 filter table 囉！不包含NAT )

例一：所有的來自 lo 這個介面的封包，都予以接受   
[root@test root]# **iptables -A INPUT -i lo -j ACCEPT**  
# 注意一下，因為 -d, --dport, -s, --sport 等等參數都沒有設定，這表示：   
# 不論封包來自何處或去到哪裡，只要是來自 lo 這個介面，就予以接受！   
# 這個觀念挺重要的，就是『沒有設定的規定，則表示該規定完全接受』的意思！   
# 例如這個案例當中，關於 -s, -d...等等的參數沒有規定時！   
　   
例二：來自 192.168.0.1 這個 IP 的封包都予以接受：   
[root@test root]# **iptables -A INPUT -i eth0 -p tcp -s 192.168.0.1 -j ACCEPT**  
# 新增一條規則，只要是來自於 192.168.0.1 的封包，不論他要去哪裡，   
# 使用的是那個協定 (port) 主機都會予以接受的意思～   
　   
例三：來自 192.168.1.0 這個 C Class 的網域的任何一部電腦，就予以接受！   
[root@test root]# **iptables -A INPUT -i eth0 -p tcp -s 192.168.1.0/24 -j ACCEPT**  
# 這個是網域的寫法喔！稍微注意一下的是，在範例二當中我們僅針對一個 IP ，   
# 至於這個範例當中，則是針對整個網域來開放吶！而網域的寫法可以是：   
# 192.168.1.0/24 也可以是 192.168.1.0/255.255.255.0 都能夠接受喔！   
　   
例四：來自 192.168.1.25 的封包都給他丟棄去！   
[root@test root]# **iptables -A INPUT -i eth0 -p tcp -s 192.168.1.25 -j DROP**

例五：來自 192.168.0.24 這個 IP 的封包，想要到我的 137,138,139 埠口時，都接受   
[root@test root]# **iptables -A INPUT -i eth0 -p tcp -s 192.168.0.24/24 --dport 137:139 -j ACCEPT**

**參數說明：**

[root@test root]# iptables [-t filter] [-AI INPUT,OUTPUT,FORWARD] \   
> [-io interface] [-p tcp,udp,icmp,all] [-s IP/network] [--sport ports]  \   
> [-d IP/network] [--dport ports] -j [ACCEPT,DROP]   
   
-A 　：新增加一條規則，該規則增加在最後面，例如原本已經有四條規則，   
　　 　使用 -A 就可以加上第五條規則！   
-I　 ：插入一條規則，如果沒有設定規則順序，預設是插入變成第一條規則，   
　　 　例如原本有四條規則，使用 -I 則該規則變成第一條，而原本四條變成 2~5   
　INPUT　：規則設定為 filter table 的 INPUT 鏈   
　OUTPUT ：規則設定為 filter table 的 OUTPUT 鏈   
　FORWARD：規則設定為 filter table 的 FORWARD 鏈   
　   
-i　　　  ：設定『封包進入』的網路卡介面   
-o　　　　：設定『封包流出』的網路卡介面   
　interface ：網路卡介面，例如 ppp0, eth0, eth1....   
　   
-p 　：請注意，這是小寫呦！封包的協定啦！   
　tcp ：封包為 TCP 協定的封包；   
　upd ：封包為 UDP 協定的封包；   
　icmp：封包為 ICMP 協定、   
　all ：表示為所有的封包！   
　   
-s     ：來源封包的 IP 或者是 Network ( 網域 )；   
--sport：來源封包的 port 號碼，也可以使用 port1:port2 如 21:23   
　　　　 同時通過 21,22,23 的意思   
-d     ：目標主機的 IP 或者是 Network ( 網域 )；   
--dport：目標主機的 port 號碼；   
　   
-j 　　：動作，可以接底下的動作；   
　ACCEPT ：接受該封包   
　DROP　 ：丟棄封包   
　LOG　　：將該封包的資訊記錄下來 (預設記錄到 /var/log/messages 檔案)

各個規則鏈的預設政策可獨立自主的設定，不必受其它鏈的影響。

以下練習，若目標為 DROP，則 policy 請設為 ACCEPT；若目標為 ACCEPT，則 policy 請設為 DROP，如此方可看出效果。

**定義變數**

|  |
| --- |
| FW\_IP="163.26.161.253" |

**打開核心 forward 功能**

作法如下：

|  |
| --- |
| ###-----------------------------------------------------###  # 打開 forward 功能  ###-----------------------------------------------------###  echo "1" > /proc/sys/net/ipv4/ip\_forward |

**開放某一個介面**

作法如下：

|  |
| --- |
| iptables -A INPUT -i lo -j ACCEPT  iptables -A OUTPUT -o lo -j ACCEPT  註：IPFW 或 Netfilter 的封包流向，local process 不會經過 FORWARD Chain，  因此 lo 只在 INPUT 及 OUTPUT 二個 chain 作用。  iptables -A INPUT -i eth1 -j ACCEPT  iptables -A OUTPUT -o eth1 -j ACCEPT  iptables -A FORWARD -i eth1 -j ACCEPT  iptables -A FORWARD -o eth1 -j ACCEPT |

**IP 偽裝**

使內部網路的封包經過偽裝之後，使用對外的 eth0 網卡當作代表號，對外連線。作法如下：

|  |
| --- |
| ###-----------------------------------------------------###  # 啟動內部對外轉址  ###-----------------------------------------------------###  iptables -t nat -A POSTROUTING -o eth0 -s 172.16.0.0/16 -j SNAT --to-source $FW\_IP |

上述指令意指：把 172.16.0.0/16 這個網段，偽裝成 $FW\_IP 出去。

**虛擬主機**

利用轉址、轉 port 的方式，使外部網路的封包，可以到達內部網路中的伺服主機，俗稱虛擬主機。這種方式可保護伺服主機大部份的 port 不被外界存取，只開放公開服務的通道(如 Web Server port 80)，因此安全性甚高。

作法如下：

|  |
| --- |
| ###-----------------------------------------------------###  # 啟動外部對內部轉址  ###-----------------------------------------------------###  # 凡對 $FW\_IP:80 連線者, 則轉址至 172.16.255.2:80  iptables -t nat -A PREROUTING -i eth0 -p tcp -d $FW\_IP --dport 80 -j DNAT --to-destination 172.16.255.2:80 |

**開放內部主機可以 telnet 至外部的主機**

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open 外部主機 telnet port 23  ###-----------------------------------------------------###  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 23 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 23 -d $FW\_IP --dport 1024:65535 -j ACCEPT |

**開放郵包轉遞通道**

開放任意的郵件主機送信包給你的 Mail Server，而你的 Mail Server 也可以送信包過去。

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open SMTP port 25  ###-----------------------------------------------------###  # 以下是：別人可以送信給你  iptables -A INPUT -i eth0 -p tcp -s any/0 --sport 1024:65535 -d $FW\_IP --dport 25 -j ACCEPT  iptables -A OUTPUT -o eth0 -p tcp ! --syn -s $FW\_IP --sport 25 -d any/0 --dport 1024:65535 -j ACCEPT  # 以下是：你可以送信給別人  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 25 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 25 -d $FW\_IP --dport 1024:65525 -j ACCEPT |

**開放對外離線下載信件的通道**

開放內部網路可以對外部網路的 POP3 server 取信件。

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open 對外部主機的 POP3 port 110  ###-----------------------------------------------------###  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 110 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 110 -d $FW\_IP --dport 1024:65535 -j ACCEPT |

**開放觀看網頁的通道**

開放內部網路可以觀看外部網路的網站。

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open 對外部主機的 HTTP port 80  ###-----------------------------------------------------###  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 80 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 80 -d $FW\_IP --dport 1024:65535 -j ACCEPT |

**開放查詢外部網路的 DNS 主機**

開放內部網路，可以查詢外部網路任何一台 DNS 主機。

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open DNS port 53  ###-----------------------------------------------------###  # 第一次會用 udp 封包來查詢  iptables -A OUTPUT -o eth0 -p udp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 53 -j ACCEPT  iptables -A INPUT -i eth0 -p udp -s any/0 --sport 53 -d $FW\_IP --dport 1024:65535 -j ACCEPT  # 若有錯誤，會改用 tcp 封包來查詢  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 53 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 53 -d $FW\_IP --dport 1024:65535 -j ACCEPT  # 開放這台主機上的 DNS 和外部的 DNS 主機互動查詢：使用 udp  iptables -A OUTPUT -o eth0 -p udp -s $FW\_IP --sport 53 -d any/0 --dport 53 -j ACCEPT  iptables -A INPUT -i eth0 -p udp -s any/0 --sport 53 -d $FW\_IP --dport 53 -j ACCEPT  # 開放這台主機上的 DNS 和外部的 DNS 主機互動查詢：使用 tcp  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 53 -d any/0 --dport 53 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! -y -s any/0 --sport 53 -d $FW\_IP --dport 53 -j ACCEPT |

**開放內部主機可以 ssh 至外部的主機**

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open 外部主機 ssh port 22  ###-----------------------------------------------------###  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 22 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 22 -d $FW\_IP --dport 1024:65535 -j ACCEPT  # 以下是 ssh protocol 比較不同的地方  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1020:1023 -d any/0 --dport 22 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 22 -d $FW\_IP --dport 1020:1023 -j ACCEPT |

**開放內部主機可以 ftp 至外部的主機**

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| ###-----------------------------------------------------###  # open 對外部主機 ftp port 21  ###-----------------------------------------------------###  # 以下是打開命令 channel 21  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 21 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 21 -d $FW\_IP --dport 1024:65535 -j ACCEPT  # 以下是打開資料 channel 20  iptables -A INPUT -i eth0 -p tcp -s any/0 --sport 20 -d $FW\_IP --dport 1024:65535 -j ACCEPT  iptables -A OUTPUT -o eth0 -p tcp ! --syn -s $FW\_IP --sport 1024:65535 -d any/0 --dport 20 -j ACCEPT  # 以下是打開 passive mode FTP 資料通道  iptables -A OUTPUT -o eth0 -p tcp -s $FW\_IP --sport 1024:65535 -d any/0 --dport 1024:65535 -j ACCEPT  iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 1024:65535 -d $FW\_IP --dport 1024:65535 -j ACCEPT |

**開放 ping**

可以對外 ping 任何一台主機。

作法如下：(預設 policy 為 DROP)

|  |
| --- |
| iptables -A OUTPUT -o eth0 -p icmp -s $FW\_IP --icmp-type 8 -d any/0 -j ACCEPT  iptables -A INPUT -i eth0 -p icm -s any/0 --icmp-type 0 -d $FW\_IP -j ACCEPT |