iptables 練習 (NASA 課程)

EXAMPLE 1-1：

1、iptables -A INPUT -i eth0 -p icmp -s 0/0 -d 0/0 -j DROP
將由 eth0 進來的 icmp 封包都 drop 掉 (無法 ping)

2、iptables -L --line-numbers
列出所選 table 的 rule，並顯示行號 (default 為 filter table)。

3、iptables -A OUTPUT -o eth0 -p icmp -s 0/0 -d 0/0 -j DROP
將經由 eth0 出去的 icmp 封包 drop 掉 (會同 1 無法 ping，了解設定 rule 時，INPUT、OUPUT 都需要考量。)

4、iptables -F (iptables -F -t nat)
清空 rule (可加-t [table name] 來清除指定 table 的 rule)

5、iptables -P INPUT DROP、iptables -P OUTPUT DROP
將 filter table 的 INPUT、OUTPUT chain 的 default policy 改為 DROP
這樣一來，一開始就全部被阻擋，而 rule 可以設定要開放的服務。

EXAMPLE 1-2：

6、iptables -A INPUT -i eth0 -p icmp -s 0/0 -d 0/0 -j REJECT
同 1，但觀察用 REJECT 跟 DROP 之間的不同
額外練習 --reject-with icmp-host-unreachable (default：icmp-net-unreachable)
--reject-with icmp-host-prohibited

以 ping 來測試 server 回應的訊息差異。

7、iptables -I INPUT -i eth0 -p icmp -s 0/0 -d 0/0 -j LOG
練習 -I insert 插入一條新的 rule 到最上面(或指定行數)，搭配 LOG 或 ACCEPT
LOG 的話，可以觀察 message log。
ACCEPT 的話，可以了解 rule 順序的重要性 (rule 執行順序為由上至下，上方的 rule 會優先執行)。

EXAMPLE 2-1：
先 telnet www.csie.ntu.edu.tw 80，確認 80 是通的 (default policy 要記得先改回 ACCEPT)。
1、iptables -A OUTPUT -o eth0 -p tcp --sport 1024:65535 --dport 80 -j DROP
rule 下了後，再 telnet www.csie.ntu.edu.tw 80，此時就不通了。
這條 rule 的作用是將內部要連往外部 80 port 的封包 drop 掉，讓內部無法連結外部的 web 服務。
**了解為什麼 sport 是 1024:65535 ? (client 端是 random port，然後 1024 以下內定是系統服務在用的，所以會設定 port range 為 1024~65535)

更改 default policy (OUTPUT)為 drop：
iptables -P OUTPUT DROP
然後先試著 ping www.csie.ntu.edu.tw (因 OUTPUT 被 drop，所以會封包無法對外送，也無法進行 dns 解析。)

2、iptables iptables -A OUTPUT -o eth0 -p udp --dport 53 -j ACCEPT
開放內部對外部的 udp 53port 的連線。
**這邊要了解的是 rule 的設定，除了 port 號外，還必需了解每個服務所用的 protocol，針對需求來開放正確的 protocol (但注意，此時只是可以對外連線 dns，所以可以解析 domain，但 ping www.csie.ntu.edu.tw 仍然是不通的，因為 icmp 的封包並沒有開放)。

更改 output 的 policy 改為 ACCEPT
3、iptables -A INPUT -i eth0 -p icmp --icmp-type echo-request -j DROP
在阻擋 INPUT 的 echo-request 後，外部的 request 無法進來(所以 ping 不到)。

---

iptables -F 清掉後
4、iptables -A OUTPUT -o eth0 -p icmp --icmp-type echo-reply -j DROP
這次是阻擋 OUTPUT 的 echo-reply 封包，因內部的回應無法傳出去 (所以一樣 ping 不到)

**這邊要了解的是：
icmp 常用的 2 種 type
type 0 ：echo-request
type 8 ：echo-reply

EXAMPLE 3-1

1、iptables -A OUTPUT -o eth0 -p tcp -m multiport --dports 53,80 -j DROP
同時設定禁止 53、80 port 的連線。(練習 multiport 的設定)

---

iptables -P INPUT DROP
iptables -P OUTPUTDROP
先把 default policy 設為 drop

2、iptables -A INPUT -i eth0 -s 0/0 -d 0/0 -p tcp --dport 22 -j ACCEPT
設定 INPUT 的 22 accept (這時候由於 output 是 drop 的，所以還是不會通，可用 telnet 192.168.219.54 22 測試)

iptables -A OUTPUT -o eth0 -s 0/0 -d 0/0 -p tcp -m state --state ESTABLISHED -j ACCPET
設定 OUTPUT state 為 ESTABLISHED 的封包允許通過。 (此時 telnet 192.168.219.54 22 測試就會通了)
**這邊要了解的是(當 server 回應 syn/ack 時，iptables 會將 state 變更為 ESTABLISHED)

DNAT & SNAT：
1、port forwarding (本機)：
iptables -t nat -A PREROUTING -p tcp -d 192.168.254.17 --dport 2222 -j DNAT --to 192.168.254.17:22
將連至 192.168.254.17:2222 導至 192.168.254.17:22
(因為是本機的關係，所以不用另外開 ip_forward 功能。)

2、ip mapping (別台機器)：

環境建置：
另一台 csie2 環境：
eth0 : 10.20.1.2
netmask : 255.255.255.0
gw : 10.20.1.1

原本的 csie1 先多加一張網卡 eth1，然後 bind 上 ip：
eth1 10.20.1.1
netmask : 255.255.255.0

先確認 2 台 的 10.20.1.0/24 網段 ip 可以互相 ping 到。

目的：將通往 192.168.254.17 (csie1) 的封包轉往 10.20.1.2(csie2)

先啟用 ip_forward 功能

sysctl -w net.ipv4.ipv_forward=1
or
cat "1" /proc/sys/net/ipv4/ip_forward

iptables -t nat -A PREROUTING -i eth0 -d 192.168.254.17 -j DNAT --to-destination 10.20.1.2
將 INPUT 進來且目的地為 192.168.254.17 的封包，修改 destination ip 為 10.20.1.2，然後 forward 出去。
(此時還是不通的，因為當 14 的 packet 回到 17 要回去 client 時，他的 source ip 是 14，就會造成後面的 connection 問題。)

iptables -t nat -A POSTROUTING -o eth0 -s 192.168.254.14 -j SNAT --to-source 192.168.254.17
將 OUTPUT 出去且來源為 10.20.1.2 的封包，修改 source ip 為 10.20.1.2。
(做 SNAT，當 14 透過 17 出去時，會帶 17 的 ip。)

試著由 pc 端，ssh 192.168.254.17，登入後會進到 csie2。

3、將內部通往外部的封包做 NAT。
iptables -t nat -F POSTROUTING
(先清掉 nat table 的 postrouting rule)

iptables -t nat -A POSTROUTING -o eth0 -s 10.20.1.0/24 -j MASQUERADE
** 這邊應了解 SNAT 跟 MASQUERADE 的差別，SNAT 可指定 output 網卡上的 ip，MASQUERADE 會自動帶上 default ip。

---

mangle
iptabbbles -t mangle -A POSTROUTING -o eth0 -j TTL --ttl-set 1
修改由 eth0 出去的封包，將 TTL 設為 1。
**觀察一下加之前之後的 traceroute 狀況。