Esercitazione

Avete creato una cartella nella vostra home di nome (tutto MAIUSCOLO)

A4A

dove mettere in disparte gli esercizi che fate (non solo quelli di oggi)?

Prima di chiedere hai usato il comando **man** ?
...sovente 1 settimana di esperimenti possono risparmiare 1 ora di lettura...

I tools di oggi

- Manipolare le interfacce di rete (ifconfig)
- Visualizzare lo stato dello connessioni di rete (netstat)
- Sbirciare il traffico di rete (tcpdump)
- Eseguire programmi con in permessi di altri (sudo)

- Leggete le manpage dei comandi suddetti e provate a:

sudo

- Permette All'amministratore di consentire ad altri utenti di eseguire comandi con il suo privilegio
- Il file di configurazione contiene ennuple nella forma
- <chi> <daDove>=(<aCheTitolo>)<cosa>
- es: in lab

ALL ALL=(root) /sbin/iptables,/usr/sbin/tcpdump,/sbin/ifconfig,/sbin/route,/usr/bin/less /tmp/user.tmp,/usr/bin/tail -f /tmp/user.tmp

- Tutti gli utenti possono eseguire i comandi dati a nome di root
- Per eseguire un comando occorre fare

sudo <comando>

es: sudo /usr/bin/tail -f /tmp/user.tmp

Esercizi sui tool

Pensa prima di provare

- fate ls /tmp/user.tmp ke diritti ha?
- potete visualizzarne il contenuto?
- fate less /tmp/user.tmp . Cosa accade?
- prova ora sudo less /tmp/user.tmp

- Scoprite che indirizzo ip ha la vostra macchina.
- Assegnatele con **ifconfig** un **ulteriore** indirizzo nella forma 192.168.199.x x=ultima cifra del vostro ip primario
- Visualizzate le connessioni aperte tramite il comando **netstat**
- tramite **tcpdump** visualizzato tutto il traffico broadcast
- in una finestra pingate la macchina di un vostro collega e in altra finestra visualizzate con tcp il traffico di rete conseguente

Ordine delle regole (1)

Se scrivo:

```
iptables -A INPUT -p icmp -j DROP
```

e poi

```
ping localhost
```

Cosa succede?

E se aggiungo:

```
iptables -A INPUT -p icmp -j ACCEPT
```

E ancora:

```
iptables -I INPUT -p icmp -j ACCEPT
```

Pensa prima di provare

Ordine delle regole (2)

- Nel primo caso il ping non va
- Nel secondo neanche (l'accept viene dopo!)
- Nel terzo sì

Filtraggio in base all'IP

Prova a scrivere 2 regole per bloccare i ping dalla macchina del tuo vicino, ma non quelli dalla macchina stessa

Filtraggio in base all'IP (2)

Una possibile soluzione:

- `iptables -A INPUT -p icmp -s 192.168.69.101/32 -j DROP`
- `iptables -A INPUT -p icmp -j ACCEPT`

Nota l'ordine

Blocco di un solo tipo di messaggio icmp (1)

- Prova ora a scrivere una regola per fare solamente il blocco dei messaggi icmp di tipo echo-reply
- Puoi poi lanciare il comando ping localhost e vedere cosa succede con tcpdump

Blocco di un solo tipo di messaggio icmp (2)

- Una possibile soluzione può essere:

```
iptables -A INPUT -p icmp –icmp-type
  echo-reply -j DROP
```

Differenza fra DROP e REJECT

Prova a scrivere:

```
iptables -A INPUT -p tcp -s 0/0 --
  dport 22 -j DROP
```

e poi fare

```
ssh -v localhost
```.

Cosa succede?

E se faccio

```
iptables -I INPUT 1 -p tcp -s 0/0 --
  dport -j REJECT
```

cosa cambia?

Pensa prima di provare

Filtraggio attraverso il mac address (1)

Fatevi dare dal vostro vicino il mac address della sua scheda di rete (come si fa?) e scrivete una regola che blocchi i ping che vengono dalla sua macchina filtrando il mac address

Filtraggio attraverso il mac address (2)

Una possibile soluzione può essere:

```
iptables -A INPUT -m mac –mac-source
  XX:XX:XX:XX:XX:XX -j DROP
```

Log dei pacchetti (1)

- Scrivete una regola per fare il log di tutti i pacchetti di echo request che vengono fatti alla vostra macchina
- I log devono essere di livello debug (vedi man page)
- La vostra macchina è configurata in modo che i log di livello debug vengono scritti nel file /tmp/user.tmp

Log dei pacchetti (2)

Una possibile soluzione può essere:

```
iptables -A INPUT -p icmp –icmp-type
  echo-request -j LOG –log-level debug
```

Catene definite dagli utenti (1)

- Create due catene personalizzate di nome Echo_request e Echo_reply
- Nella catena di INPUT fate in modo che i messaggi icmp di tipo echo-request vengano passati alla catena Echo_Request e quelli di tipo echo-reply alla catena Echo_Reply
- Aggiungete una regola alla catena Echo_Request in modo che il pacchetto venga loggato a livello debug con prefisso 'Sono un echo request '
- Aggiungete una regola alla catena Echo_Reply in modo che il pacchetto venga loggato a livello debug con prefisso 'Sono un echo reply '

Catene definite dagli utenti (2)

Una possibile soluzione può essere:

- `iptables -N Echo_Request`
- `iptables -N Echo_Reply`
- `iptables -A INPUT -p icmp –icmp-type echo-request -j Echo_Request`
- `iptables -A INPUT -p icmp –icmp-type echo-reply -j Echo_Reply`
- `iptables -A Echo_Request -s 0/0 -j LOG –log-prefix 'Sono un echo request ' --log-level debug`
- `iptables -A Echo_Reply -s 0/0 -j LOG –log-prefix 'Sono un echo reply ' --log-level debug`

Compitini per casa

Usare iptable per:

- Fare il log tutto il traffico diretto alla porta 631 della propria macchina
- Fare il log del traffico diretto alla porta 631 della propria macchina che non sia proveniente dalla macchia stessa
- Fare 1 script che dato:
  - un file (**elenco**) contentente un insieme di macchine
  - un parametro (**durata**) espresso in secondi

consenta di far passare ping dalla propria macchina verso una qualunque delle macchine presenti in elenco per un tempo "**durata**" dalla partenza dello scritp. Scaduto il periodo blocchi l'uscita dei ping, ma consenta comunqe alle altre macchine di pingare la propria

supercompitino

- Costruire uno script che:
  - impedisca a chiunque di accedere da remoto via ssh
  - all'arrivo di un ping da una macchina(m) consenta alla macchina (m) stessa di accedere alla nostra macchina via ssh
- Modificare lo script in modo che al successivo ping la macchina smetta di accettare conenssioni ssh