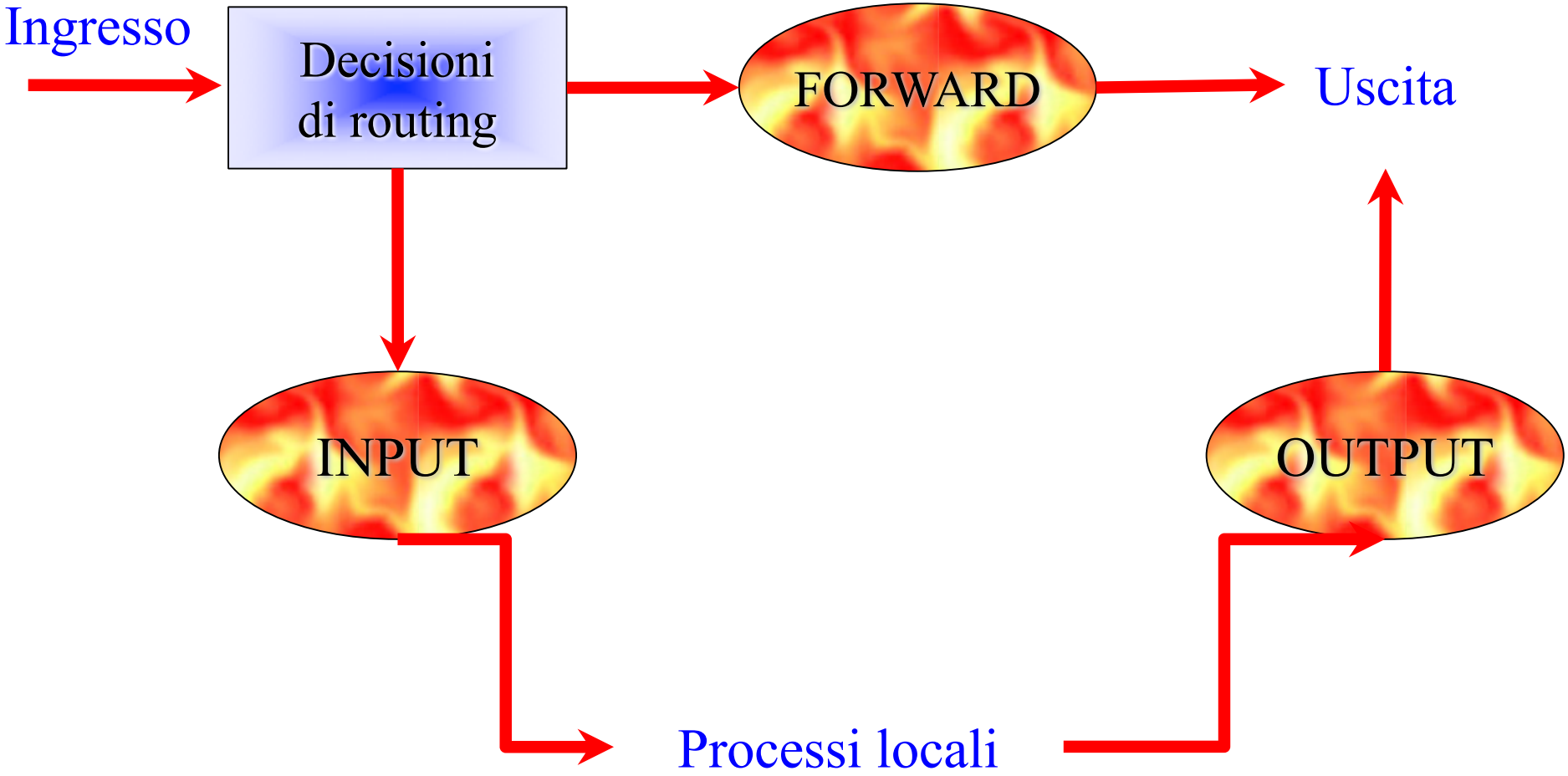
IPTABLES

Corso di Network Security
Prof. Simon Pietro Romano

Filtro dei pacchetti con Linux

- Un filtro dei pacchetti è un software che guarda gli header dei pacchetti e ne decide il destino
- Può decidere di scartare il pacchetto, lasciarlo passare o fare qualcosa di più complicato
- Il kernel di Linux contiene un'infrastruttura che consente il filtraggio dei pacchetti
- Tale infrastruttura è chiamata *netfilter*
- Il tool *iptables* dialoga con il kernel e gli indica quali pacchetti filtrare, inserendo e rimuovendo regole dalla tabella di filtraggio del kernel

Come i pacchetti attraversano i filtri



Come i pacchetti attraversano i filtri

- INPUT, OUTPUT e FORWARD sono tre *liste di regole*, chiamate *catene*, presenti nella tabella *filter*
- Quando un pacchetto arriva ad uno di questi blocchi, la catena corrispondente lo esamina per deciderne il destino
- Se la catena dice di scartare (DROP) il pacchetto, questo sarà ucciso lì, se la catena dice di accettarlo (ACCEPT), il pacchetto potrà continuare a percorrere il diagramma
- Una catena è una lista di regole. Ogni regola dice 'se l'header del pacchetto appare in questo modo, allora ecco cosa bisogna fare'

Come esaminare le regole di una catena

1) Se il pacchetto non soddisfa una certa regola, allora sarà consultata la regola successiva
2) Se non ci sono più regole da consultare, allora il kernel utilizza la *tattica* (policy) della catena
3) In un sistema per la sicurezza, questa tattica in genere indica al kernel di scartare il pacchetto

Usare iptables

- Il tool iptables mette a disposizione alcune operazioni utili per gestire intere catene:
  - Crea una nuova catena (-N)
  - Cancella una catena vuota che non è l'obiettivo di alcuna regola (-X)
  - Cambia la tattica di una delle catene preesistenti (-P)
  - Elenca le regole presenti in una catena (-L)
  - Svuota una catena delle sue regole (-F)
  - Azzera i contatori dei pacchetti e dei byte di tutte le regole di una catena (-Z)
- Le catene preesistenti INPUT, FORWARD e OUTPUT non possono essere cancellate

Usare iptables

- Ci sono poi diversi modi per manipolare le regole di una catena:
  - Appendi una nuova regola alla catena (-A)
  - Inserisci una nuova regola in una determinata posizione della catena (-I)
  - Sostituisci una regola presente in una certa posizione della catena (-R)
  - Cancella una regola presente in una certa posizione della catena (-D)
  - Cancella la prima regola di una catena (-D)

Una semplice regola

iptables -A INPUT -s 143.225.229.60 -p icmp -j DROP

- Appende alla catena INPUT la regola che dice di scartare tutti i pacchetti icmp provenienti da 143.225.229.60
- Per cancellare una regola si può fornire come parametro il numero associato alla regola
  - es. iptables -D INPUT 5

  cancella dalla catena INPUT la quinta regola
- Per evitare di contare le regole (e sbagliarsi), si può immettere lo stesso comando usato per aggiungere la regola utilizzando l'opzione -D
  - es. iptables -D INPUT -s 143.225.229.60 -p icmp -j DROP

Regole

- Come detto, una regola specifica un insieme di condizioni che il pacchetto deve soddisfare, e che cosa fare se le soddisfa (obiettivo)
- Vedremo nelle slide seguenti quali condizioni è possibile specificare
- In seguito, vedremo come specificare gli obiettivi

Specificare gli indirizzi IP

- Gli indirizzi IP sorgente ('-s', '--source', '--src') e destinazione ('-d','--destination', '--dst') possono essere specificati in 4 modi diversi
  - Nome completo es. www.linuxhq.com
  - Indirizzo IP es. 143.225.229.130
  - Indirizzi IP es. 143.225.229.0/24
  - Indirizzi IP es. 143.225.229.0/255.255.255.0

- Nota: molte opzioni, incluse '-s' e '-d', possono avere gli argomenti preceduti da '!' per indicare gli indirizzi NON uguali a quello indicato
  - es. -s ! localhost indica qualsiasi pacchetto non proveniente da localhost

Specificare un protocollo e un'interfaccia

- Il protocollo può essere specificato usando l'opzione '-p' (o '--protocol')
- Il protocollo può essere un numero o un nome, ad es. tcp, UDP (maiuscolo o minuscolo non fa differenza)
- Le opzioni '-i' (o '--in-interface') e '-o' (o '--out-interface') servono a specificare il nome di una interfaccia
- I pacchetti che attraversano la catena INPUT non hanno un'interfaccia di output, perciò qualsiasi regola che usa '-o' in questa catena non troverà mai una corrispondenza
- Analogamente, i pacchetti che attraversano la catena OUTPUT non hanno un'interfaccia di input
- Solo i pacchetti che attraversano la catena FORWARD hanno sia un'interfaccia di input che una di output

Specificare i frammenti

- Il problema dei frammenti è che il frammento iniziale contiene gli header completi (IP + TCP, UDP o ICMP) da esaminare, mentre quelli successivi contengono solo un sottoinsieme degli header (IP)
- Il comportamento è il seguente: qualsiasi regola che richieda informazioni in realtà non presenti, NON sarà soddisfatta
- es. -p TCP --source-port www

  è soddisfatta solo dal primo frammento
- Si può comunque indicare una regola specifica per il secondo e i successivi frammenti usando l'opzione -f

Estensioni di iptables

- iptables è estendibile, ossia sia il kernel che il tool iptables possono essere estesi per fornire nuove caratteristiche
- Alcune di queste estensioni sono standard
- Le estensioni sono di due tipi: nuovi confronti e nuovi obiettivi
- Alcuni protocolli (TCP, UDP e ICMP) offrono nuovi test
- È possibile specificare questi nuovi test sulla linea di comando dopo l'opzione '-p', che provvederà a caricare l'estensione

Estensioni TCP

- --tcp-flags

permette di filtrare in base ai flag di TCP (la prima stringa di flag indica i flag che vuoi esaminare, la seconda indica quelli che dovrebbero essere settati)

es. iptables -A INPUT -p tcp --tcp-flags ALL SYN,ACK -j DENY

- --syn

equivale a --tcp-flags SYN,RST,ACK SYN

- --source-port e --destination-port

specifica una porta o un intervallo di porte TCP:

  - Nome o numero
  - 5000:6000
  - 5000:
  - :5000

Estensioni TCP

- A volte si può voler permettere connessioni verso un server esterno ma si vuole impedire che tale server stabilisca una connessione con il nostro PC
- Si può pensare allora di bloccare tutti i pacchetti TCP provenienti dal server
- Sfortunatamente, le connessioni TCP hanno bisogno di scambiare pacchetti in entrambe le direzioni
- La soluzione consiste nel bloccare solo i pacchetti usati per richiedere una connessione
- Tali pacchetti hanno il flag SYN impostato e i flag FIN e ACK azzerati. Quindi:

  -p TCP -s 192.168.1.1 --syn

Estensioni UDP e ICMP

- UDP
  - È possibile utilizzare le opzioni --sport e --dport
- ICMP
  - Fornisce una nuova opzione (--icmp-type) che consente di filtrare i diversi tipi di pacchetti ICMP
  - È possibile specificare il nome o il codice numerico
  - Con -p icmp --help si ottiene un elenco dei tipi di pacchetto ICMP

Altre estensioni

- Le altre estensioni (se installate) possono essere invocate utilizzando l'opzione -m
- Estensione mac
  - Questo modulo viene caricato specificando '-m mac' e fornisce l'opzione '--mac-source' che consente di filtrare i pacchetti in base all'indirizzo ethernet sorgente (in notazione esadecimale)
- Estensione multiport
  - Specificando '-m multiport' si possono specificare fino a 15 porti (separati da ',') utilizzando le opzioni:
    - --source-ports
    - --destination-ports

Altre estensioni

- Estensione state
  - Specificando '-m state' si può utilizzare l'opzione '--state' che indica una serie di stati da confrontare:
    - NEW (un pkt che crea una nuova connessione)
    - ESTABLISHED (un pkt appartenente ad una connessione esistente)
    - RELATED (un pkt che è relativo, ma che non fa parte, di una connessione esistente, es. un errore ICMP o un pacchetto che tenta una connessione ftp data)
    - INVALID (un pkt che non si riesce ad identificare)

Altre estensioni

- Estensione limit
  - Questo modulo deve essere specificato con '-m limit'. È usato per restringere il numero di confronti. Il funzionamento è analogo a quello del *token bucket*. Argomenti:
  - --limit

  specifica la media massima di confronti permessi (equivalente al *token rate*) al secondo (es. '5/sec'), al minuto, all'ora o al giorno
  - --limit-burst

  è equivalente al *bucket size*
  - Protezione dal syn-flood:

```
iptables -A FORWARD -p tcp --syn -m limit --limit 1/s -j ACCEPT
```

  - Protezione dal ping of death:

```
iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT
```

Specificare gli obiettivi

- L'obiettivo (target) di una regola dice cosa fare dei pacchetti che soddisfano i criteri di selezione
- Ci sono due semplici obiettivi già disponibili: DROP e ACCEPT
- Ci sono altri due tipi di obiettivi:
  - Le catene create dall'utente
  - Le estensioni

Catene create dall'utente

- È possibile creare nuove catene, che si aggiungono a quelle già disponibili (INPUT, FORWARD e OUTPUT)
- Per convenzione, i nomi delle catene create dall'utente sono in minuscolo
- Quando un pacchetto soddisfa una regola che ha per obiettivo una catena creata dall'utente, il pacchetto comincia ad attraversare le regole di quest'ultima
- Se la catena termina e il destino del pacchetto non è stato deciso allora si torna nuovamente alla catena corrente

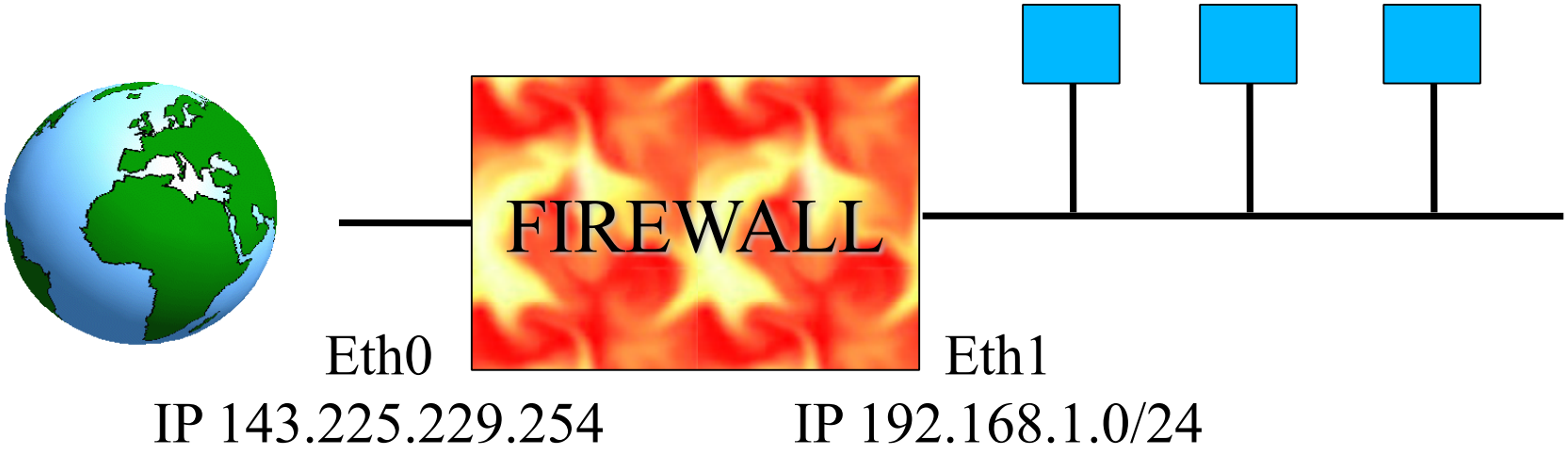
Obiettivi speciali già disponibili

- RETURN
  - Ha lo stesso effetto di quando si arriva alla fine di una catena: se la regola corrente appartiene a una catena predefinita, allora viene applicata la tattica della catena. Se appartiene ad una catena definita dall'utente, si prosegue con la catena precedente
- QUEUE
  - I pacchetti che soddisfano i criteri della regola vengono accodati per elaborazioni 'userspace'
  - È necessario un gestore delle code che si occupi del meccanismo di passaggio tra kernel e userspace
  - L'applicazione userspace riceve ed elabora tali pacchetti

Nuovi obiettivi

- LOG
  - Questo modulo permette la registrazione da parte del kernel dei pacchetti specificati
  - Le informazioni sui pacchetti sono inviate al demone syslogd per essere registrate nel file /var/log/messages
  - Fornisce le opzioni --log-level e --log-prefix
  - iptables continua l'elaborazione con la regola successiva
- REJECT
  - Questo modulo ha lo stesso effetto di DROP, però in più viene inviato in risposta un messaggio di errore ICMP di tipo 'port unreachable'

Esercizio



Esercizio

- Creare uno script che, dopo aver eliminato tutte le regole e le catene definite dall'utente, configuri il firewall in modo che:

1. La tattica di default di tutte le catene sia DROP
2. Tutti i pacchetti che arrivano dall'esterno ed hanno un indirizzo sorgente appartenente alla rete interna (spoofing) siano bloccati
3. I pacchetti ICMP diretti all'indirizzo broadcast della rete interna (smurf) siano bloccati
4. Tutti i pacchetti TCP appartenenti a connessioni esistenti stabilite con i server www e smtp della rete interna siano lasciati passare

Esercizio

5. Le richieste di connessione provenienti dall'esterno verso i server www e smtp siano consentite
6. Le richieste di connessione dei PC della rete interna verso i server smtp, www, ftp e irc del mondo esterno siano consentite
7. Le richieste verso (le risposte da) server DNS esterni siano consentite
8. I rimanenti pacchetti (che verranno scartati) siano sottoposti a “logging”

Una possibile soluzione

```
iptables -F
iptables -X
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
iptables -A FORWARD -s 192.168.1.0/24 -i eth0 -j DROP
iptables -A FORWARD -p icmp -i eth0 -d 192.168.1.255 -j DROP
iptables -A FORWARD -m multiport -p tcp -d 192.168.1.0/24 --
    destination-ports www,smtp -m state --state ESTABLISHED,
    RELATED -j ACCEPT
iptables -A FORWARD -m multiport -p tcp -s 192.168.1.0/24 --
    source-ports www,smtp -m state --state ESTABLISHED,
    RELATED -j ACCEPT
```

Una possibile soluzione

```
iptables -A FORWARD -m multiport -p tcp -i eth0 -d
   192.168.1.0/24      --destination-ports www,smtp -m state      --
   state NEW -j ACCEPT

iptables -A FORWARD -m multiport -p tcp -i eth1                    --
   destination-ports www,smtp,ftp,irc  -m state --state NEW       -j
   ACCEPT

iptables -A FORWARD -p udp -i eth1 --destination-port domain   -j
   ACCEPT

iptables -A FORWARD -p udp -i eth0 --source-port domain         -j
   ACCEPT

iptables -A OUTPUT -j LOG

iptables -A INPUT -j LOG

iptables -A FORWARD -j LOG
```