FIREWALLING, CONFIGURAZIONI E NETFILTER

Corso di Sicurezza e Gestione delle reti

LEONARDO MACCARI: LEONARDO.MACCARI@UNIFI.IT
LART - LABORATORIO DI RETI E TELECOMUNICAZIONI
DIPARTIMENTO DI ELETTRONICA E TELECOMUNICAZIONI

TOC

Firewall

Un firewall è un apparato software o hardware configurato per ammetterre, abbattere o veicolare (proxy firewall) connessioni tra due aree di rete con differente livello di fiducia.

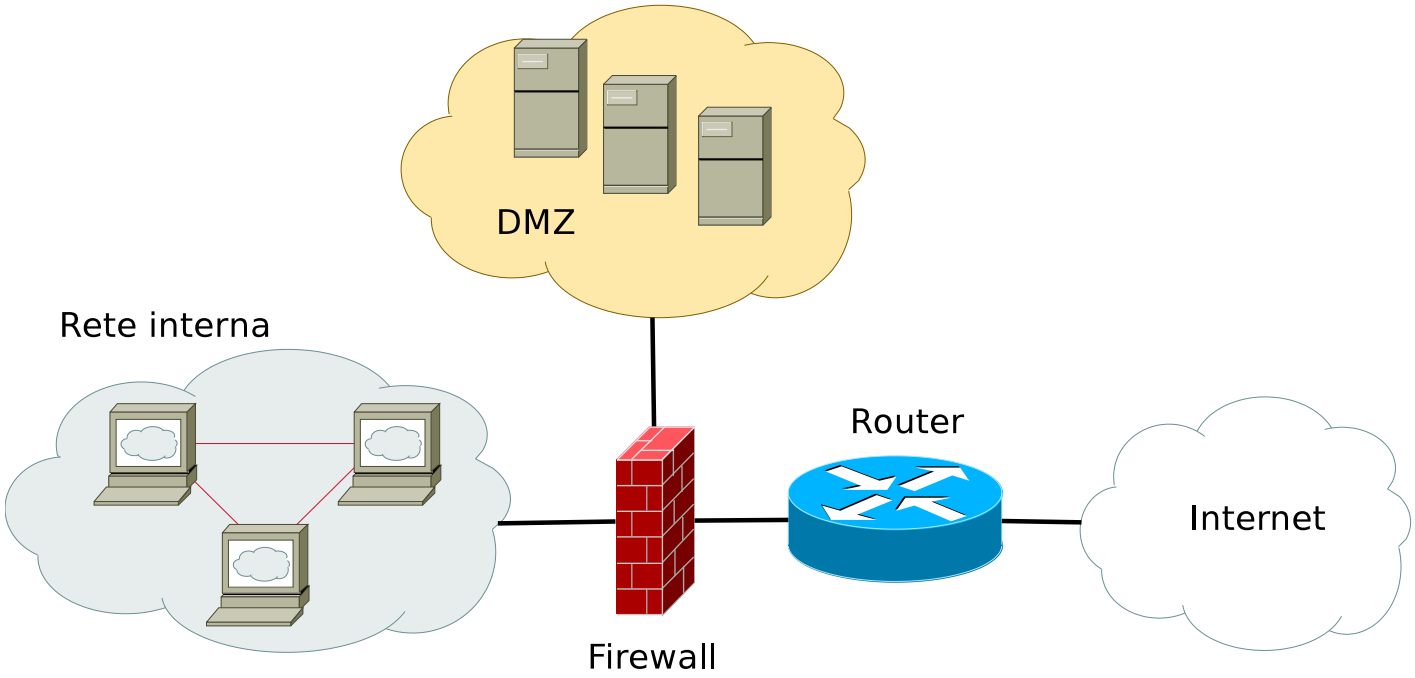
- Esempio: un firewall perimetrale viene normalmente posto su un gateway per separare la rete locale (alto livello di fiducia) da intenet (livello di fiducia minimo)
- Lo scopo finale del firewall è di offrire un'interfaccia configurabile tra due segmenti di rete con diversi livelli di fiducia. L'interfaccia deve essere configurabile attraverso security policy basate su due principi:
  - Least privilege
  - Separation of duties
- La configurazione di un firewall richiede profonda conoscenza dei protocolli di rete e di network security, un errore nella configurazione può rendere inutile il suo utilizzo.

Evoluzione dei firewall:

- Packet filter: ogni pacchetto passa attraverso il firewall e per ognuno di questi si prende una decisione separata. La decisione presa all'istante t non è condizionata dalle scelte fatte per i pacchetti precedenti.
- Stateful firewall: nel firewall vengono implementate delle macchine a stati per prendere decisioni più complesse. Ad esempio, non accettare un pacchetto di ACK TCP se non è stato prima ricevuto un pacchetto di SYN.
- Application layer firewall: i firewall operano normalmente a livello di rete, o in casi specifici a livello di collegamento, livelli in cui il formato dei pacchetti è definito e non può cambiare. Gli application firewall leggono le informazioni del payload del pacchetto per decidere quale applicazioni possono passare. Richiedono una complessità maggiore e quindi maggiori risorse computazionali.

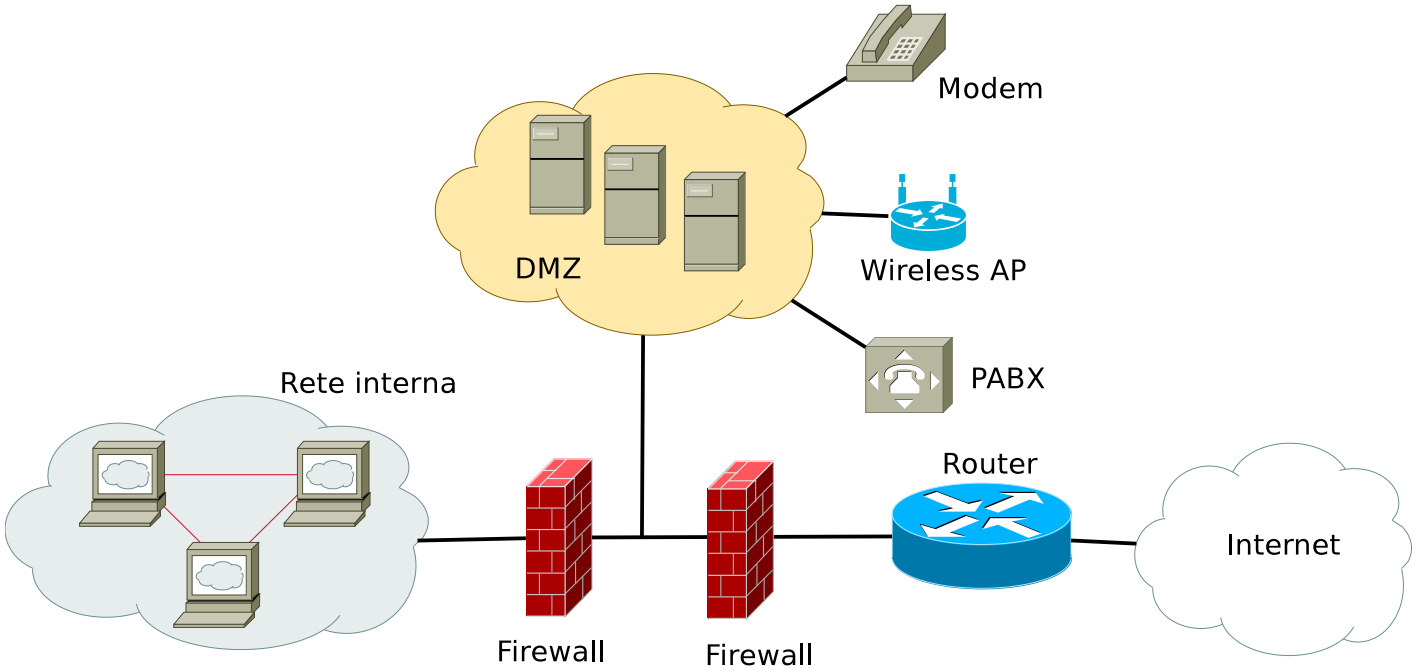
Piazzamento del firewall:

- Il firewall viene utilizzato per separare aree distinte della rete. Una configurazione tipica è quella in cui il firewall separa due segmenti di rete:
  - una rete interna (corporate) in cui risiedono le postazioni degli utenti, i server di dati e i database, quindi il segmento che contiene le informazioni più importanti per l'attività e che deve essere più protetta.
  - una rete accessibile dall'esterno, sui cui risiedono i server web, di posta e DNS, che sono a diretto contatto con Internet quindi a maggiore rischio. Questa zona contiene dati accessibili dall'esterno, quindi in linea di principio di minore valore e con minori restrizioni per l'accesso (nell'ottica di chi vede la rete da fuori). Si definisce DeMilitarized Zone (DMZ).


DMZ
Rete interna
Router
Internet
Firewall

Piazzamento del firewall:

- Una seconda configurazione prevede di aggiungere un ulteriore firewall in modo da avere due elementi di difesa prima di arrivare alla rete corporate. La configurazione è più robusta perchè:
  - un attaccante dovrebbe bucare due firewall prima di arrivare alla rete corporate (i firewall utilizzeranno software o hardware diversi, offrendo ridondanza)
  - La DMZ è separata anche dall'interno verso l'esterno, con lo stesso principio.
  - É più facile separare il traffico, quindi altri tipi di connessioni verso l'esterno che possono essere considerati meno sicuri si possono inserire nella DMZ.


Modem
DMZ
Wireless AP
PABX
Rete interna
Router
Internet
Firewall
Firewall

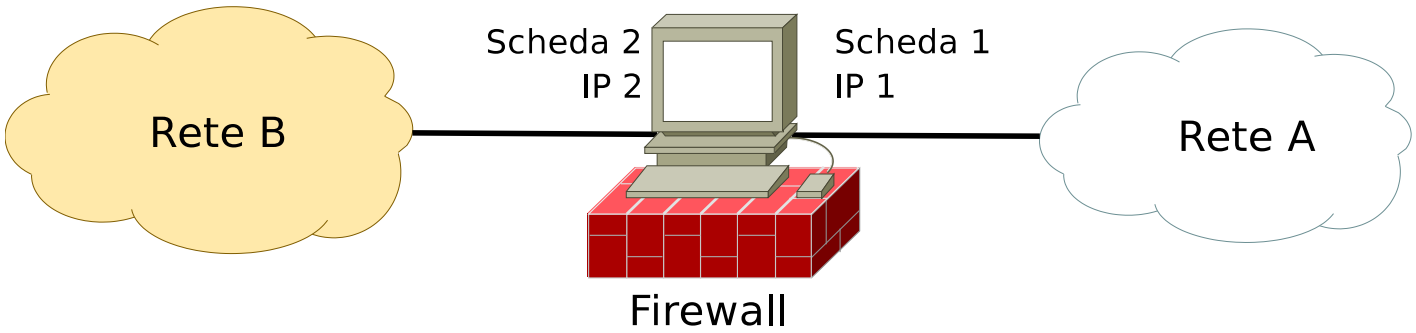
Netfilter/Iptables

- Netfilter è il framework inserito nel kernel di GNU/Linux che permette di effettuare filtraggio dei pacchetti su un firewall software.
- Netfilter lavora in *kernel space* ovvero nel nucleo del sistema operativo e mette a disposizione degli *hook*, ovvero dei punti di aggancio in cui i pacchetti possono essere filtrati durante il percorso all'interno del firewall.
- Iptables è uno strumento che permette di inserire, cancellare ed organizzare le regole di scarto, ovvero le regole secondo cui i pacchetti vengono filtrati nel kernel. Un esempio di regola:
  - iptables -t filter -D INPUT –dport 80 -j ACCEPT
    - -t filter: tabella
    - -D input: catena
    - –dport 80: criterio di match della regola
    - -j ACCEPT: target
    - traduzione: accetta i pacchetti in arrivo sulla porta 80

Catene e Tabelle

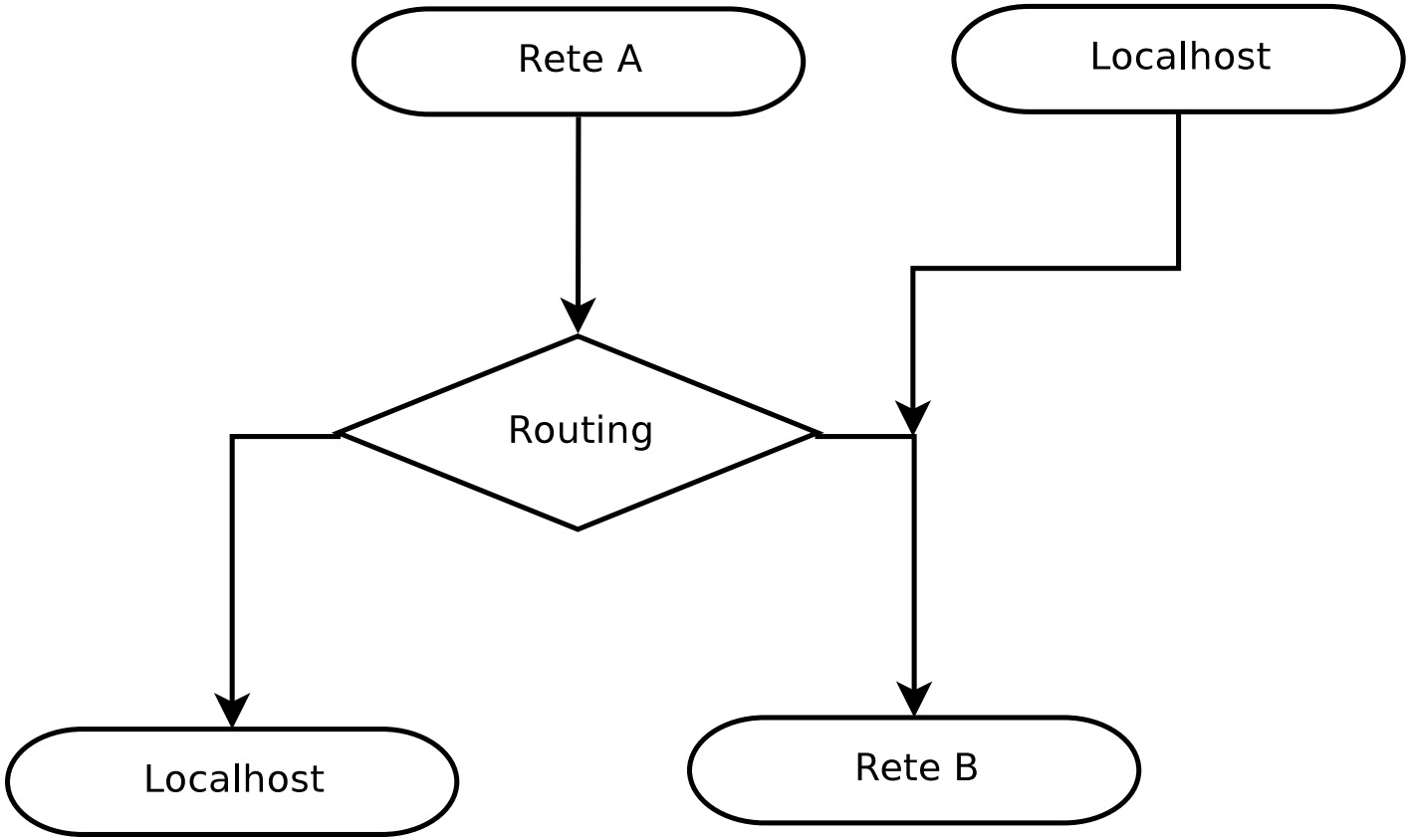
Le regole sono organizzate in catene e tabelle:

- Una catena identifica il punto all'interno del percorso nel kernel in cui avviene il filtraggio.
- Una tabella associa una funzione alla regola.
- Che cosa vuol dire?



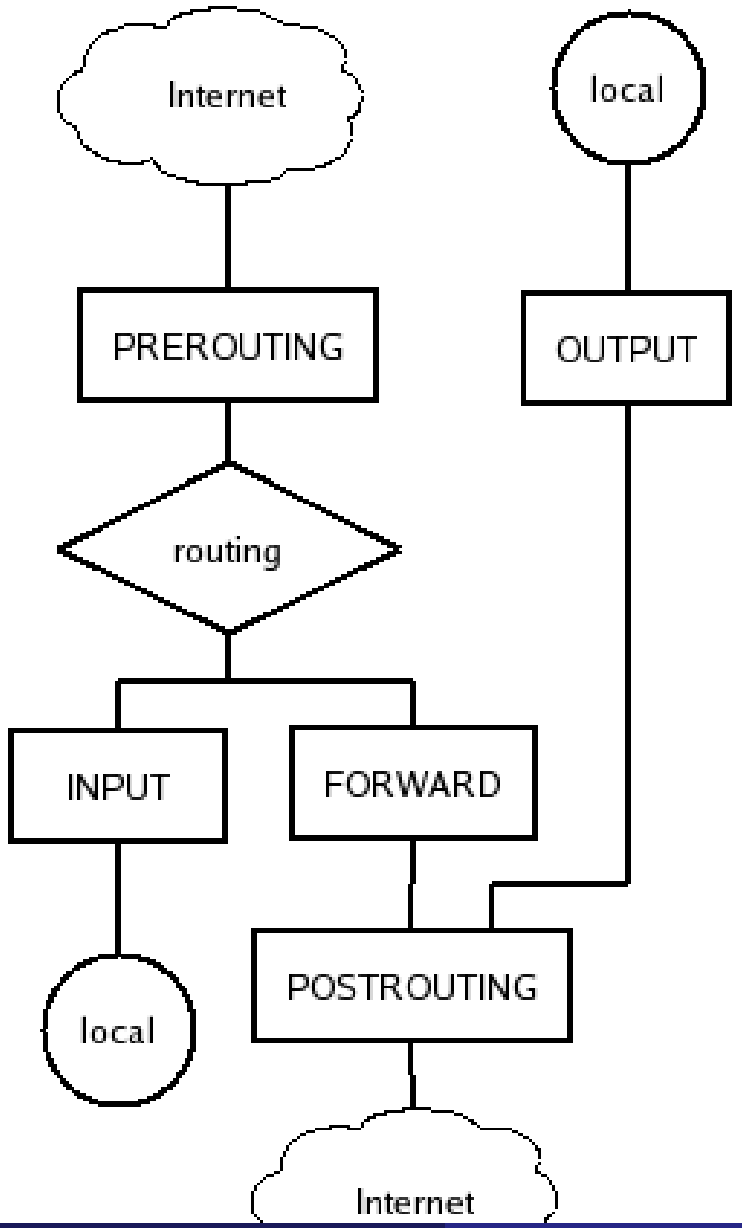
- Un firewall è un host a tutti gli effetti, con almeno due schede di rete, ognuna delle quali possiede un indirizzo IP.
- I pacchetti possono arrivare su una delle schede, essere filtrati ed essere reinviati sull'altra (forwarding)
- Se un pacchetto ricevuto dalla scheda 1 è diretto all'IP 1, il pacchetto viene elaborato in locale, e non c'e' forwarding
- Il firewall può generare dei pacchetti, che vengono inviati all'esterno verso altri IP su una delle due schede.

Schema logico del firewall



- Dove si devono/possono filtrare i pacchetti?

Netfilter: le catene



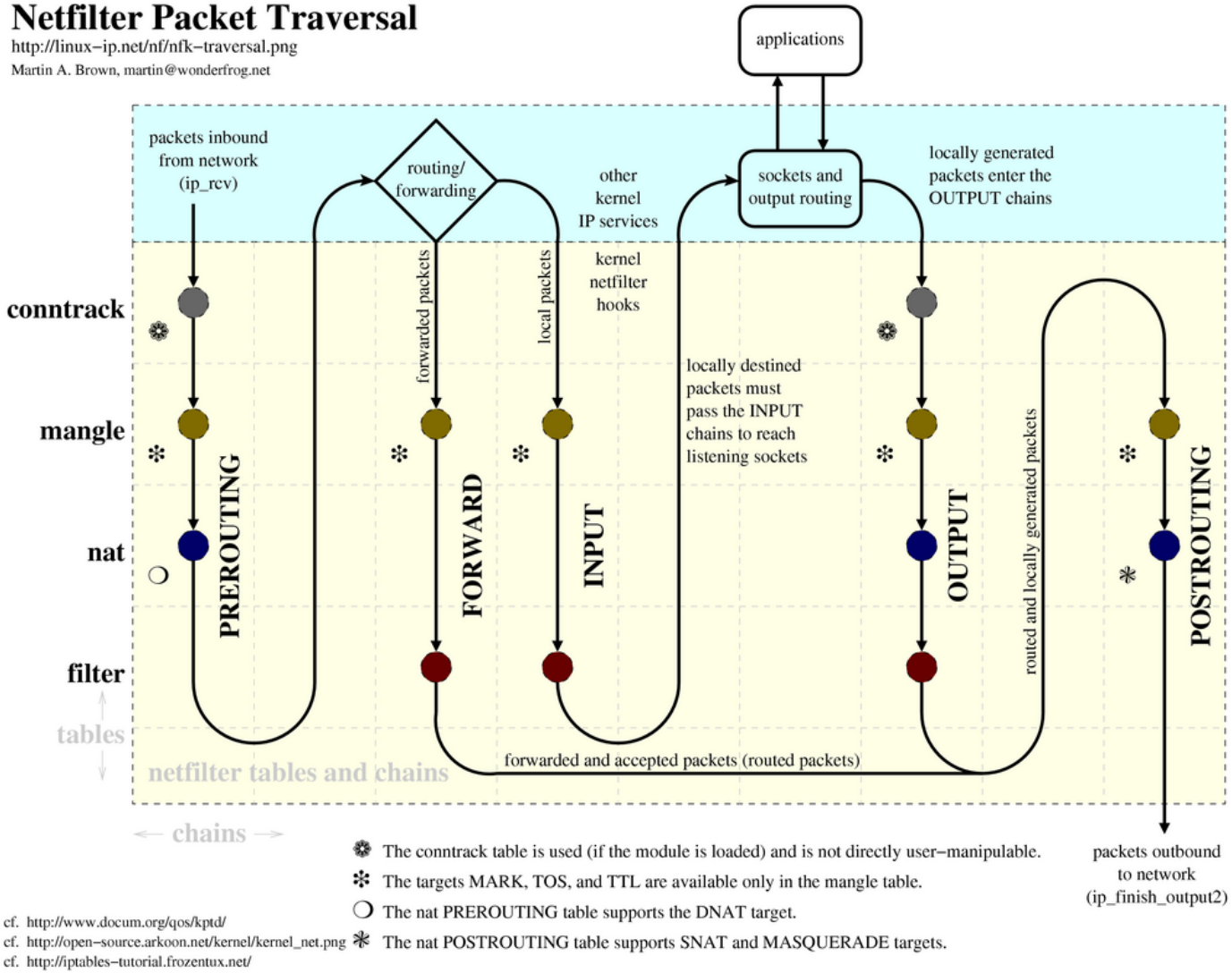
- Prerouting: tutti i pacchetti in ingresso al firewall
- Postrouting: tutti i pacchetti in uscita dal firewall
- Output: pacchetti generati dal firewall in uscita
- Input: pacchetti in ingresso al firewall diretti al firewall
- Forward: pacchetti in ingresso al firewall ma provenienti dall'esterno

Che vuol dire filtrare?

- Scartare (Drop)
- Lasciare passare (Accept)
- Modificare (Mangle)
- Lasciare passare ma riportare un messaggio nei log (Log)
- ...

- Inoltre Netfilter/IPtables è stateful, quindi ci deve essere un modulo che ricostruisce il flusso di pacchetti correlati, ad esempio frammenti dello stesso pacchetto IP (Conntrack)
- Per distinguere gruppi di azioni simili, le regole vengono divise in tabelle: ovvero raggruppamenti di regole che svolgono la stessa funzione.
  - Conntrack
  - Mangle
  - NAT
  - Filter
- All'interno di ogni catena vengono richiamate regole appartenenti a tabelle diverse.

Lo schema completo



La tabella di NAT

NAT table: Network address translation, serve a modificare i cmpi di indirizzo IP dentro agli header del pacchetto. I target possibili sono:

- DNAT: destination address translation, si cambia l'indirizzo IP destinazione. Viene utilizzato dai firewall di frontiera per distribuire il carico su una rete con più server.
  - iptables -t nat -I POSTROUTING -s 192.168.1.12 -j SNAT –to-source 150.217.5.123
- SNAT: source address translation. si cambia l'indirizzo IP sorgente. Viene utilizzato per mascherare una rete privata, di indirizzi non routabili dietro ad un indirizzo pubblico.
  - iptables -t nat -I PREROUTING -d 150.217.5.123 -j DNAT –to-destination 192.168.1.12

La tabella di Filter

Filter table: serve a operare il vero filtraggio dei pacchetti, decide quali passano e quali vengono bloccati. I target possibili sono

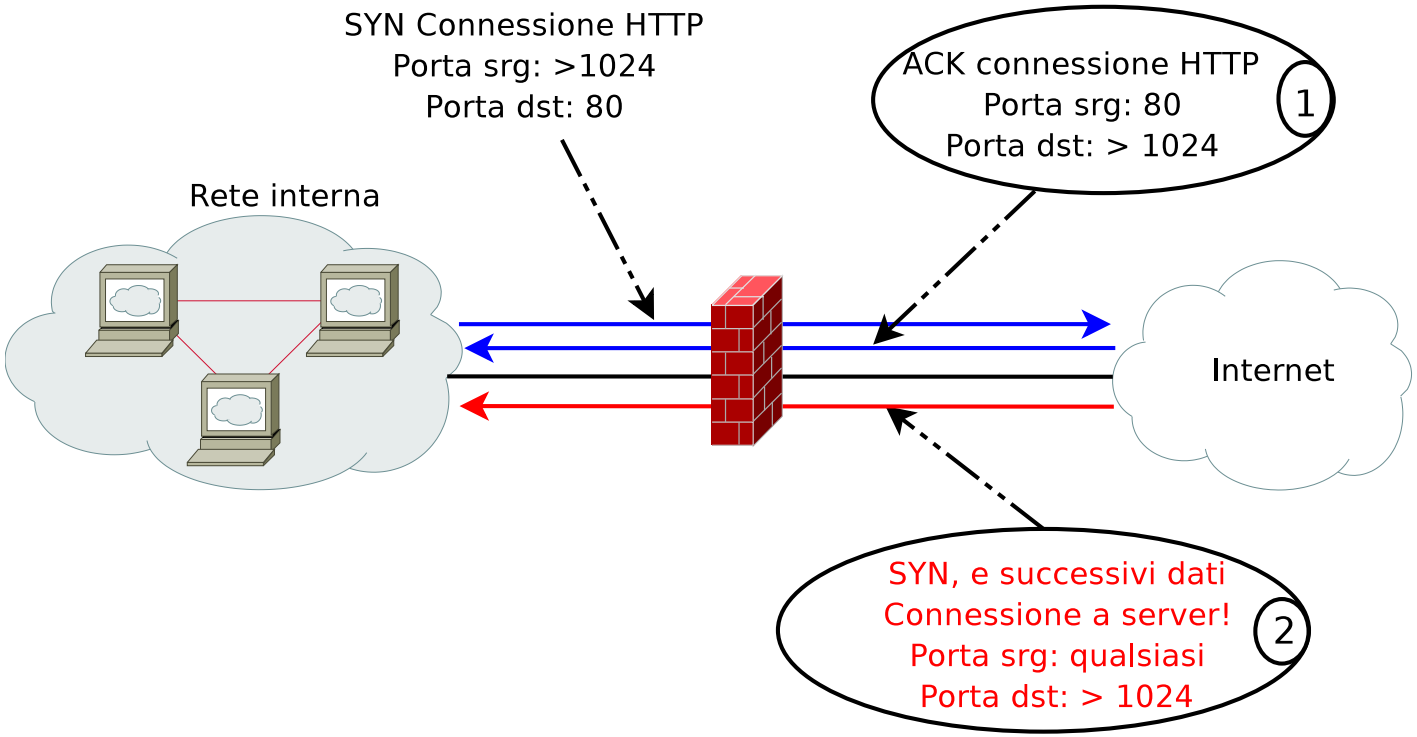
- Drop: Il pacchetto viene scartato senza dare risposta al mittente.
- Reject: il pacchetto viene scaratato inviando a destinazione una risposta di reset
- Accept: il pacchetto continua il suo percorso all'interno del kernel
- Log: il pacchetto genera un log (su schermo, su file...)

Il connection tracking

- Il modulo di Conntrack svolge alcune funzioni fondamentali nell'azione di filtraggio, ma che vanno utilizzate con attenzione o si rischia di saturare le risorse della macchina. Lo scopo è quello di mettere in relazione pacchetti diversi, secondo il funzionamento di una macchina a stati, per individuare:
  - frammenti che costituiscono lo stesso pacchetto IP
  - pacchetti che fanno parte della stessa connessione
  - pacchetti che fanno parte di connessioni distinte ma relazionate tra loro (ad esempio connessioni FTP)

Il connection tracking

- Esempio: in un firewall che protegge una rete privata, normalmente non si vogliono permettere connessioni dall'esterno verso le porte alte:

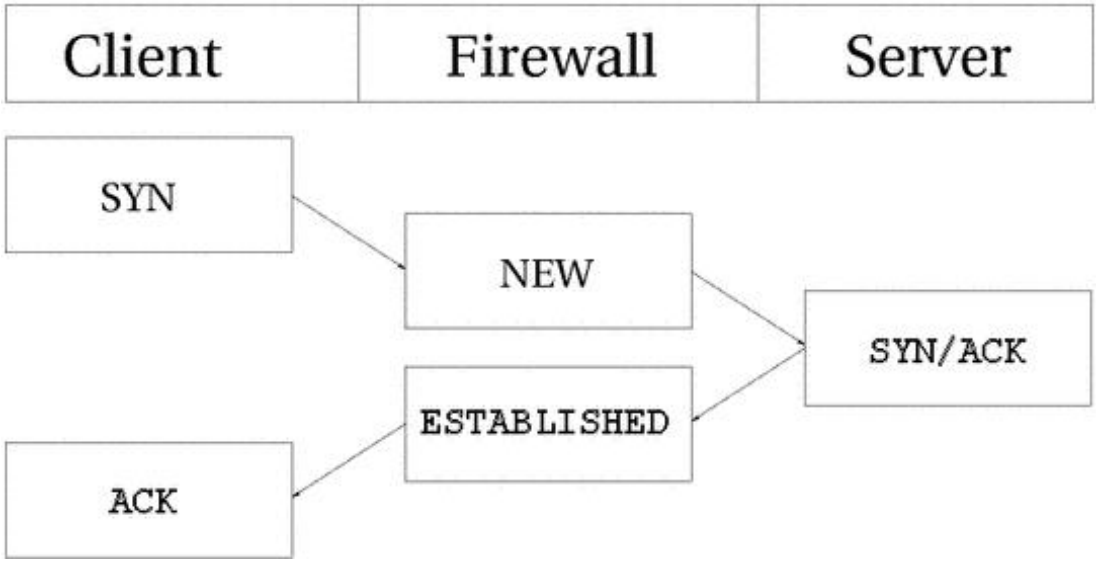


- Come posso distinguere i pacchetti 1 e 2?

Il connection tracking

- Usare il tipo (SYN) non è conveniente, il problema si riproporrebbe con altri protocolli, ad es. UDP. Esiste una differenza fondamentale:
  - Il pacchetto 1 viene ricevuto dopo aver inviato un pacchetto in uscita
  - Il pacchetto 2 invece inizia la connessione
- Il modulo conntrack tiene traccia di queste associazioni. Ogni pacchetto (di qualsiasi tipo, UDP, TCP) viene inserito in una *connessione* che può trovarsi in 4 stati:
  - NEW: il kernel ha visto passare pacchetti in una sola direziona
  - ESTABLISHED: il kernel ha vistro traffico in entrambe le direzioni
  - INVALID: nessuna delle precedenti, si è verificato un errore
  - RELATED: per usi specifici, il pacchetto appartiene ad una connessione in qualche modo relazionata ad una già ESTABLISHED

Il connection tracking: state machine



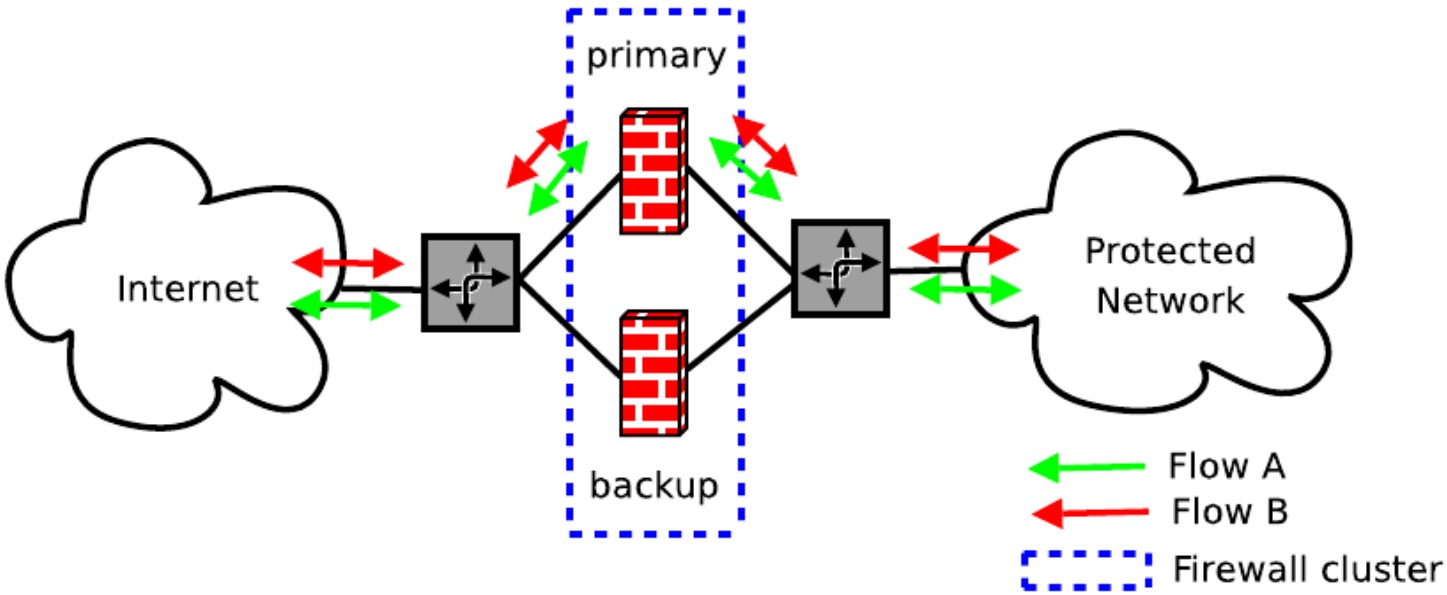
- iptabels -A INPUT -j ACCEPT -p tcp -m state –state ESTABLISHED
- iptabels -A OUTPUT -j ACCEPT -p tcp -m state –state NEW, ESTABLISHED
- iptables -P INPUT DROP;

Fault Tolerance and Load Balancing

- Il Firewall è normalmente un punto in ingresso e di uscita dalla rete e può costituire un collo di bottiglia.
- In reti che sono soggette ad alti volumi di traffico è importante condividere il carico tra più firewall per avere prestazioni migliori e ad evere procedure di backup.
  - Backup cold swap: ci sono due firewall, uno spento uguale al primo, quando si rompe il primo si accende il secondo
  - Backup hot swap: il secondo firewall è sempre acceso ed entra in funzione quando il primo smette di funzionare.
- Si possono usare configurazioni diverse.

Primary-Backup configuration

- Il gateway smista il traffico ai due firewall, il primary possiede un indirizzo virtuale (VIP) che è quello che vedono le applicazioni dall'esterno.
- Il Backup è generalmente inattivo.
- Si usa un protocollo di *heartbeat* (VRRP, HSRP ecc. . .) per controllare lo stato del server primario, quando questo subisce un guasto il VIP viene assegnato al server di backup.



Primary-Backup configuration

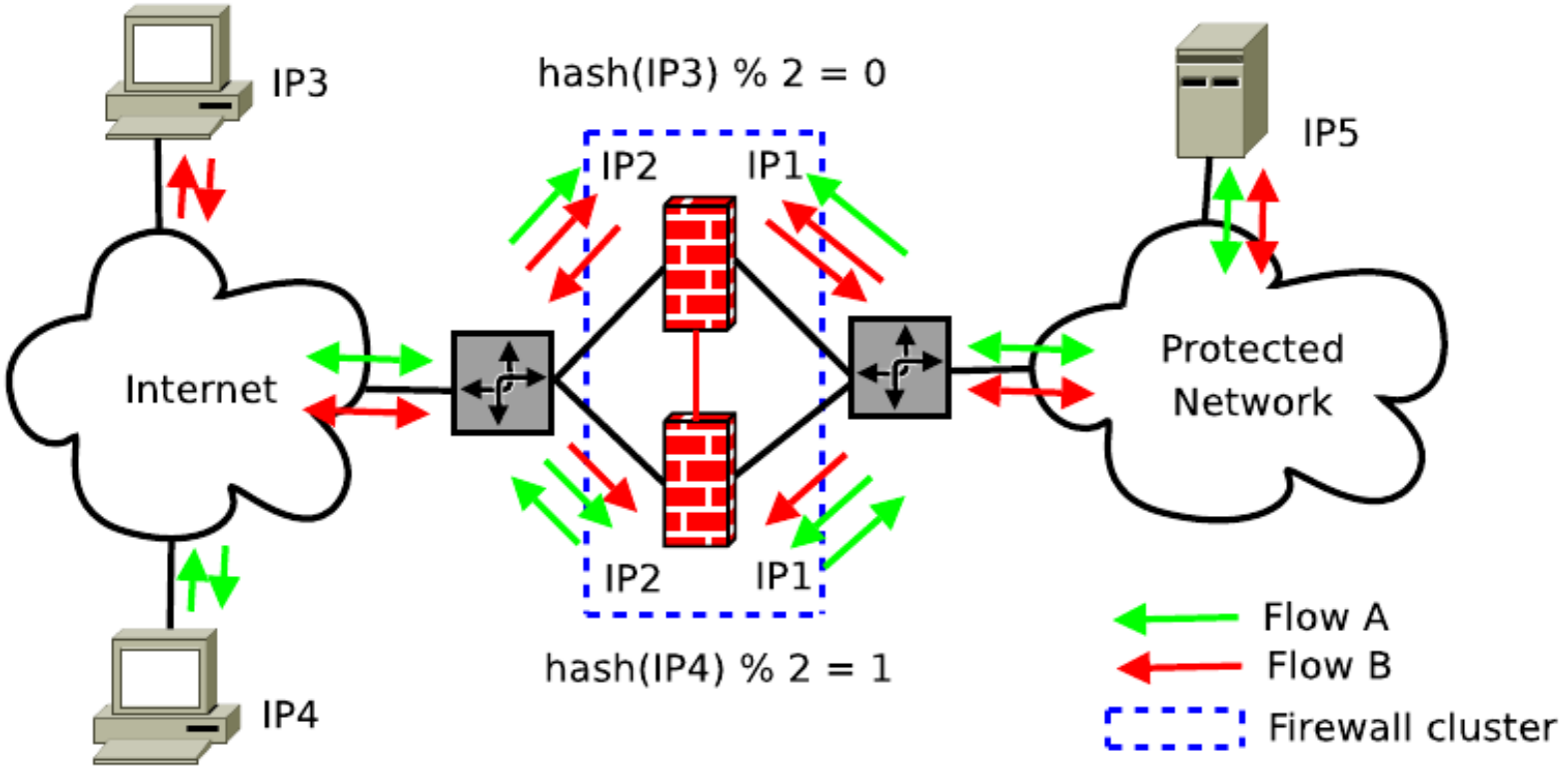
- Non c'e' load balancing.
- Nel momento del guasto le connessioni cadono tutte.
- C'e' spreco di risorse, perchè una macchina non fa niente.

Multi-primary multi-path firewall cluster

- E’ uguale al caso precedente ma prima della coppia di firewall c’e’ un load balancer che distribuisce i flussi di traffico su entrambi i firewall.
- C’e’ load balancing.
- Se uno dei firewall si guasta cadono tutte le sue connessioni
- Il problema della ridondanza si sposta sul load balancer

Multi-primary hash-based stateful firewall-clusters

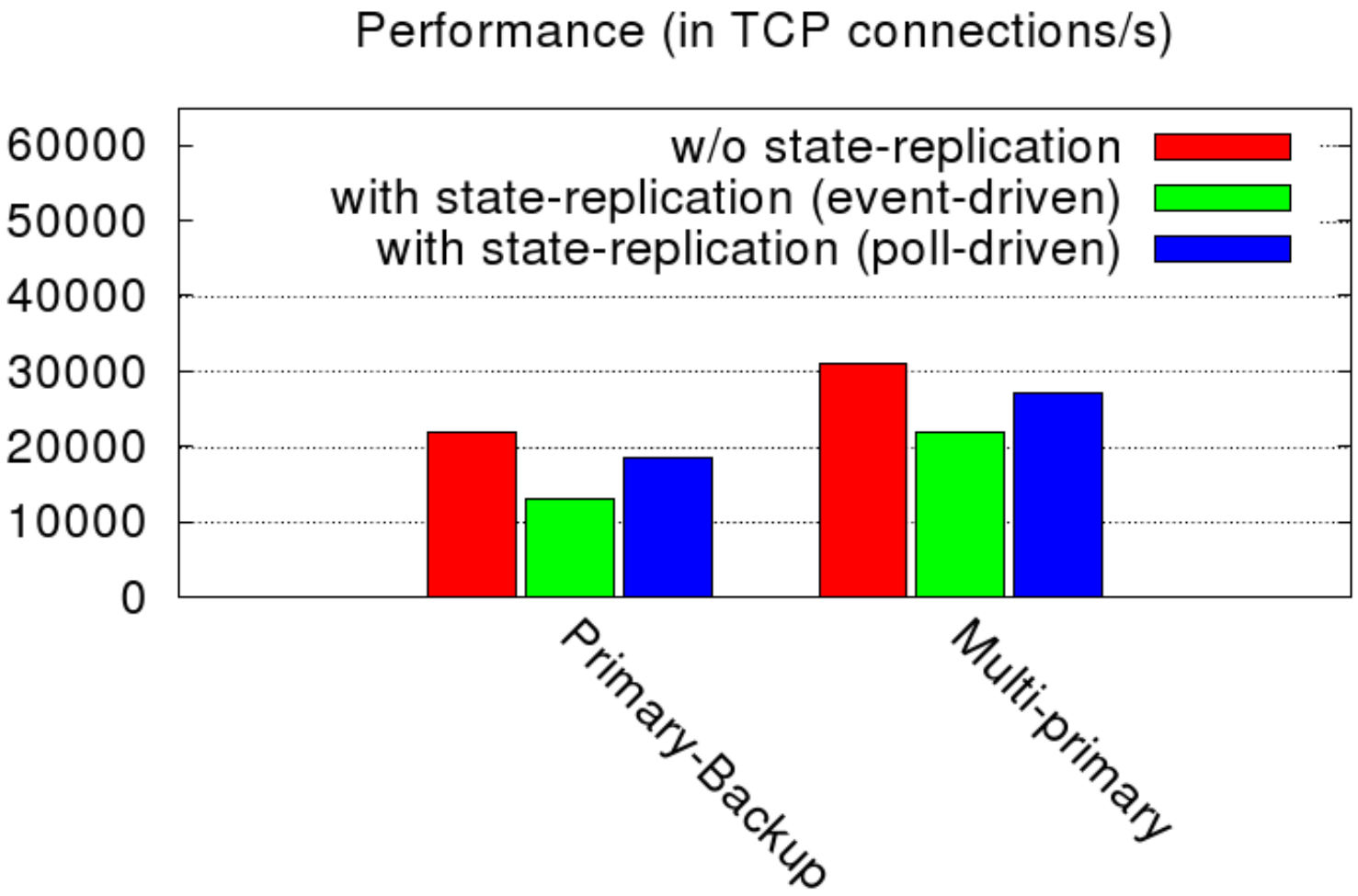
- Non c'e' load balancer. Ogni firewall ha un ID numerico (0,1...) e valuta una connessione in ingresso attraverso una tupla T = $IP_s, IP_d, Port_s, Port_d, Protocol$.
- Per ogni tupla: se hash(T) % 2 == ID allora filtra, altrimenti ignora
- In questo modo i firewall si distribuiscono il traffico autonomamente.
- C'e' comunque bisogno di un heartbeat per reagire nelle situazioni di guasto.



State replication

- In tutte queste situazioni, quando un firewall si guasta si perdono le connessioni attive in quel momento.
- Per evitare questa conseguenza è necessario che nel momento in cui una connessione cambia stato su un firewall, queso cambio venga replicato nell'altro. Si può fare con due politiche:
  - Su base evento (ad ogni cambio si comunica al backup)
  - Update periodici
- Queste due strategie hanno performance diverse in termini di affidabilità ma anche costi computazionali differenti

State replication



L7 Filtering

Un amministratore di rete può voler filtrare traffico di livello applicazione per vari motivi:

- Log e analisi del traffico. Volete sapere quale è il tipo di traffico che passa nella vostra rete per dimensionare efficacemente i link e gli apparati
- Traffic shaping. Volete dare priorità ad alcuni flussi piuttosto che ad altri.
- Blocco di alcuni protocolli. Volete evitare che alcuni tipi di traffico passino sulla vostra rete.

Si filtra a livello 7 quando non è sufficiente utilizzare il numero di porta sorgente e destinazione per capire che tipo di traffico si sta analizzando.

L7 Filtering

Filtrare protocolli di livello 7 è molto difficile :

- Esistono meccanismi interni dei protocolli che rendono difficile collegare connessioni diverse alla stessa sessione (FTP, SIP . . .)
- Esistono protocolli che intenzionalmente cercano di offuscare il loro tipo, in modo da non essere distinguibili.
- Esistono protocolli cifrati.

Quindi ogni filtro deve essere modellato sull'applicazione specifica e può avere una macchina a stati molto complessa.

L7 Filtering - difficoltà

- Implementare macchine a stati complicate per filtrare gigabit di traffico è computazionalmente molto pesante. E' necessario avere macchine dedicate con potenza sufficiente.
- I protocolli. E' possibile che da un giorno al successivo un filtro smetta di funzionare causando perdita di performance (falsi negativi) o blocco di connessioni legittime (falsi positivi).
- Un algoritmo di pattern matching implementato in software ha gli stessi problemi di sicurezza di altri applicativi di livello 7. Cosa che generalmente è più difficile per firewall di livello più basso.

L7 Filtering - security

Vulnerabilità note:

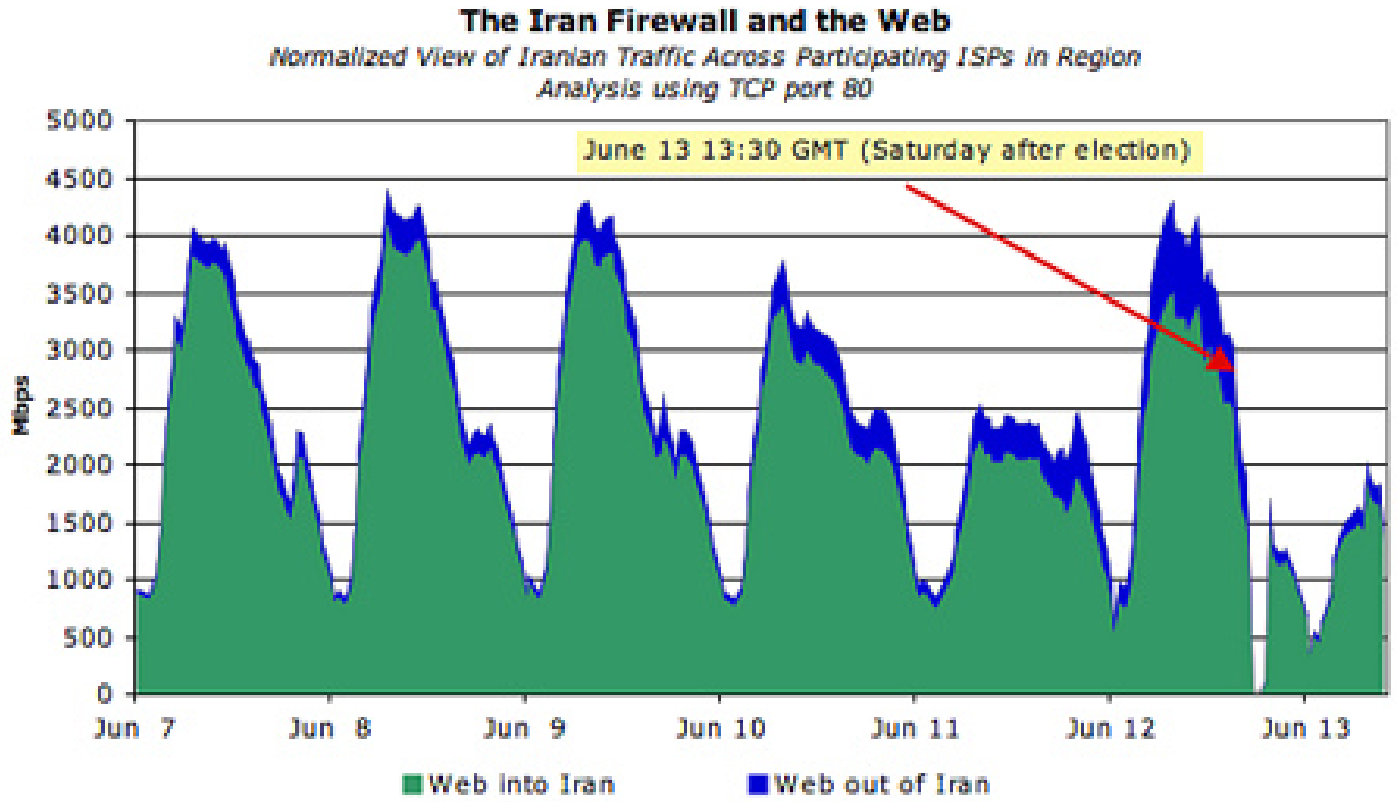
- **Snort RPC Preprocessing Vulnerability:** Researchers at Internet Security Systems (ISS) discovered a remotely exploitable buffer overflow in the Snort stream4 preprocessor module [. . .] Remote attackers may exploit the buffer overflow condition to run arbitrary code on a Snort sensor.
- **Trend Micro InterScan VirusWall Remote Overflow:** An implementation flaw in the InterScan VirusWall SMTP gateway allows a remote attacker to execute code with the privileges of the daemon.
- **Microsoft ISA Server 2000 H.323 Filter:** Remote Buffer Overflow Vulnerability. The H.323 filter used by Microsoft ISA Server 2000 is prone to remote buffer overflow vulnerability.
- **Cisco SIP Fixup Denial of Service (DoS):** The Cisco PIX Firewall may reset when receiving fragmented SIP INVITE messages.

L7 Filtering - neutrality

- Quando la banda a disposizione non è sufficiente, o si aumenta la banda o si fa traffic shaping. Nel secondo caso si decide di rendere prioritari alcuni traffici rispetto ad altri.
- Chi offre servizi quindi diventa arbitro di quale tipo di traffico è prioritario, ovvero la rete di trasporto non è più neutrale.
- La perdita di neutralità viene spesso vista come un tentativo di censurare alcuni contenuti dalla rete.

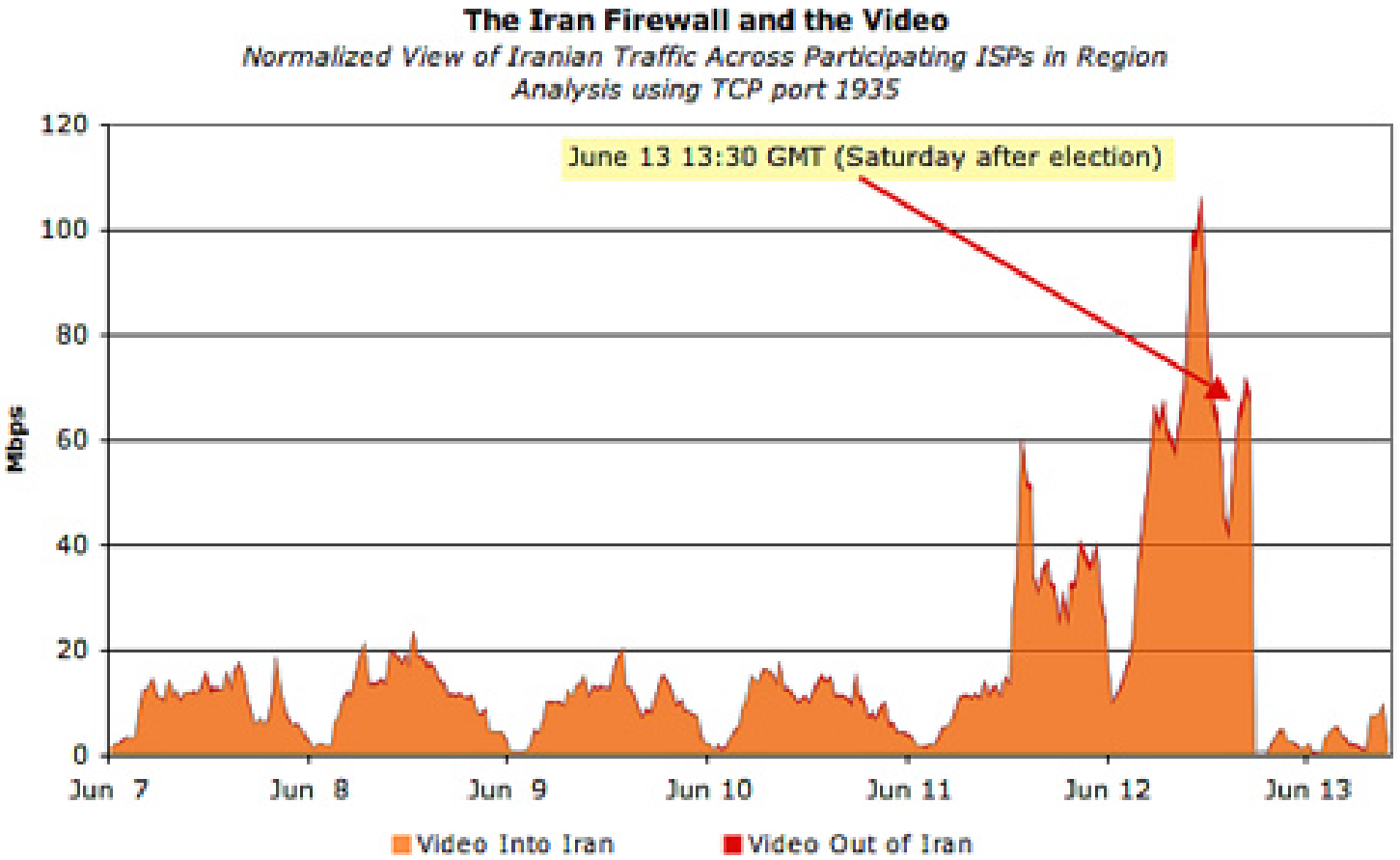
L7 Filtering - extreme example

13 Giugno 2009: elezioni in Iran. Le immagini della repressione cominciano a fare il giro del mondo[1].

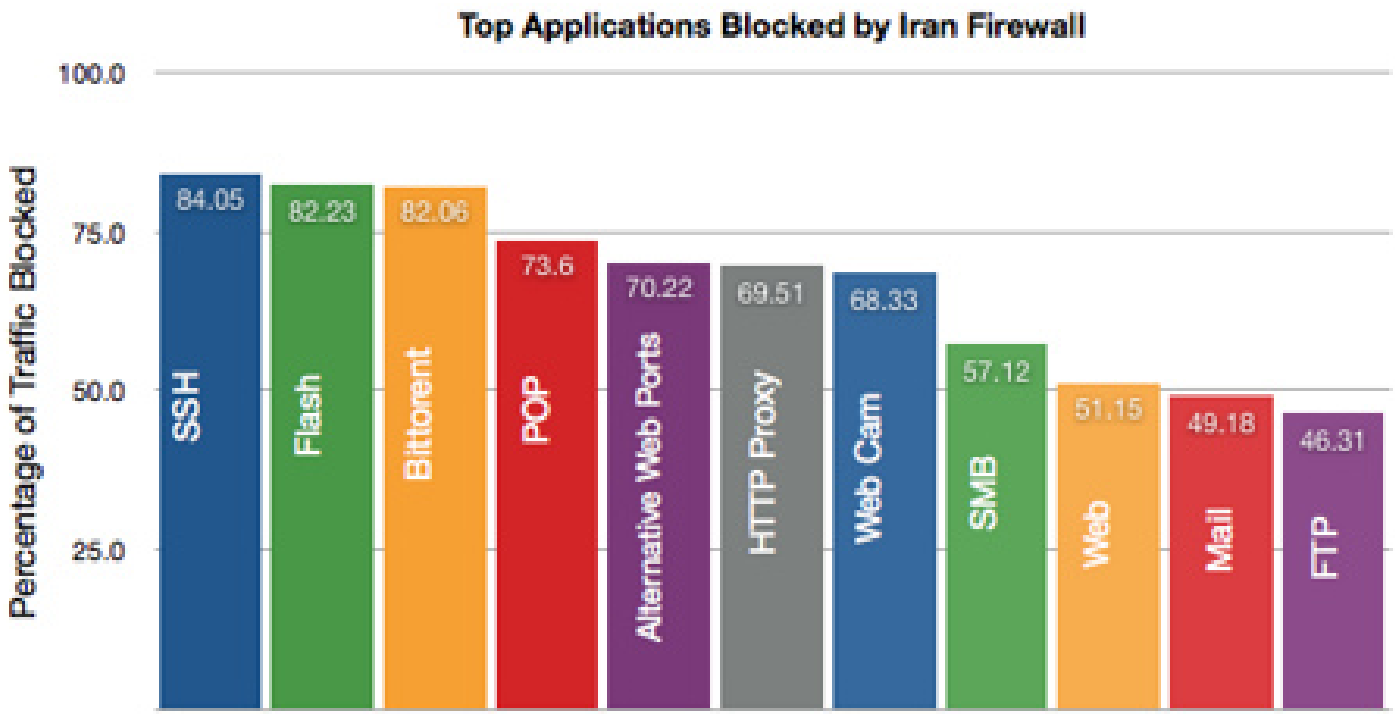


[1] images from http://asert.arbornetworks.com/2009/06/a-deeper-look-at-the-iranian-firewall/

L7 Filtering - extreme example



L7 Filtering - extreme example



L7 Filtering - overselling

- Generalmente i provider vendono servizi che in teoria non possono garantire.
- Se tutti gli utenti di un provider utilizzassero le risorse contemporaneamente non ce ne sarebbero a sufficienza.
- I provider contano sul fatto che l'utilizzo dia eterogeneo e diversificato nel tempo.
- Questo approccio non va d'accordo con i protocolli P2P (file-sharing, skype...) riescono a sfruttare risorse anche nei momenti in cui l'utente non fa niente.
- Questo, insieme ad una certa avversione nata negli ultimi anni contro il P2P, ha prodotto molta attenzione sui prodotti di Deep-Packet-Inspection (ovvero l7 filtering)

Riferimenti

- Netfilter internals e connection tracking: http://people.netfilter.org/pablo/docs/login.pdf
- fault-tolerant firewalls: *Demystifying cluster-based fault-tolerant Firewalls* P. Neira, R.M. Gasca L. Lefevre, IEEE internet computing nov 2009 (non ancora pubblicato, chiedi al tuo professore :-))
- The Perils of Deep Packet Inspection, Dr. Thomas Porter http://www.securityfocus.com/infocus/1817