IPTABLES - NETFILTER

Sistemi molto comodi quando si vuole mascherare alle applicazioni il funzionamento della rete [sostanzialmente facendo un NAT in locale].

Immaginiamo di avere un HOST LINUX nella cui parte kernel è implementato NETFILTER, che permette di utilizzare delle regole che vengono applicate ai pacchetti di rete che o attraversano il mio host, o sono creati da applicazioni che girano sul mio host.

⇒ qualunque pacchetto che si trova su quell'host può essere selezionato, e può subire modifiche.

IPTABLES: comando che può essere lanciato [⇒ si trova a livello applicazione] e tramite esso indicare a Netfilter quali regole deve applicare e a quali pacchetti.

⇓

Preso un gateway avente un interfaccia sulla rete privata, ed un interfaccia sulla rete pubblica, è possibile trasformarlo in un NAT tramite il controllo, ed eventualmente il delete, dei pacchetti.

Tramite iptables e netfilter è possibile selezionare un pacchetto, girarlo ad una applicazione locale che lo manipola e successivamente lo rispedisce nelle code del kernel ⇒ per es: potrei intercettare i pacchetti dell'applicazione SKYPE, passarli alla mia applicazione locale, modificarli in un qualche modo [o anche scartarli] per poi rispedirli a SKYPE come se nulla fosse.

NATURALMENTE PER FARE CIÒ SONO NECESSARI PRIVILEGI DI AMMINISTRATORE DI SISTEMA!

SCHEMA GESTIONE PACCHETTI DI UN HOST DA PARTE DEL SISTEMA NETFILTER:

FASE 1: Prerouting

Il sistema riceve un pacchetto ⇒ bisogna capire a chi è indirizzato tale pacchetto, se ad un processo locale o ad un altro host [⇒ la nostra macchina funge da relay];

[Se il pacchetto è diretto ad un processo locale all'host ⇒ viene inserito nella coda di INPUT del processo, avendo eventualmente la possibilità di modificarlo prima di passarlo al processo; una volta uscito dal processo, ovvero inserito nella coda di output del processo, il pacchetto può essere sempre modificato.

A questo punto il pacchetto può essere rispedito ad un altro processo locale, oppure indirizzato verso l'esterno della rete. *

→ Se il pacchetto non è diretto ad un processo locale, ovvero è diretto all'esterno entra nella fase di FORWARDING, durante la quale può comunque essere manipolato. *

⇒

A questo (*) punto vi è un ulteriore fase facoltativa: devo capire prima chi è il next hop a cui inoltrarlo, cioè devo capire qual'è l'interfaccia di rete attraverso la quale spedirlo

↳ una volta fatto tutto quanto si entra nella fase di ROUTING, dove posso sempre modificarlo prima di re-inserirlo nella rete.

N.B. Durante la fase di ROUTING posso effettuare la modifica dell'indirizzo mittente [⇒ NATTARE il pacchetto]