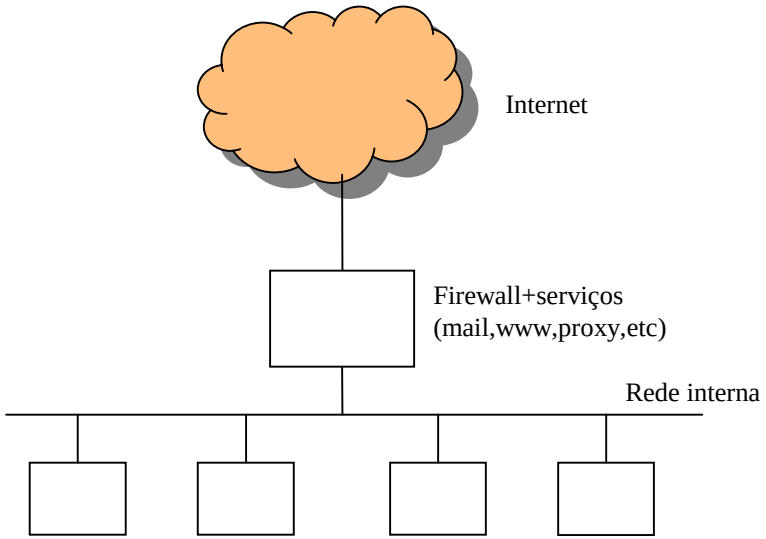
Capítulo 4

TCP/IP FIREWALLS.

- O que é uma firewall? É um router entre uma rede privada e uma rede pública que filtra o tráfego com base num conjunto de regras.

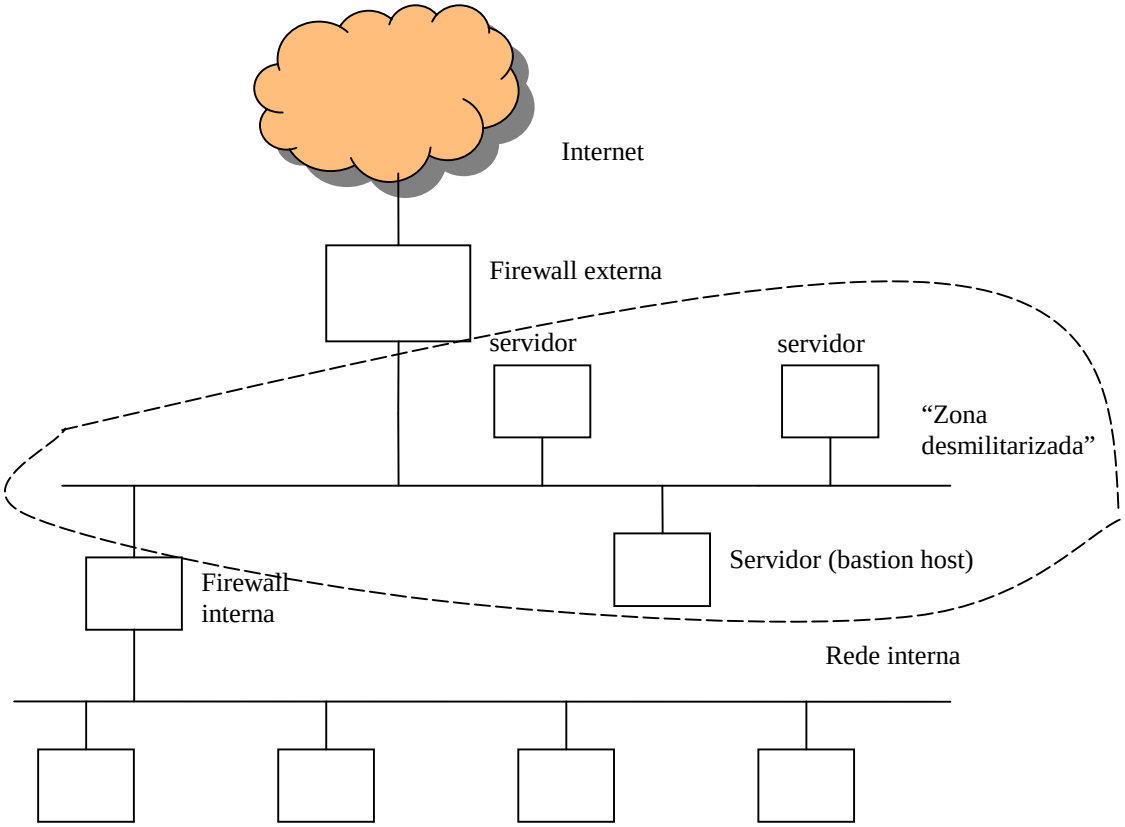
Arquitecturas de redes com firewall

Simples:



Arquitecturas de redes com firewall (2)

mais usual:



A firewall filtra o tráfego ao nível das camadas de ligação e/ou rede e/ou transporte:

- Interface de rede de entrada ou saída
- IP endereço de origem ou IP endereço de destino
- protocolo (TCP, UDP, ICMP, etc)
- porta de origem ou destino (TCP ou UDP)
- flags TCP (SYN/ACK/FIN etc)

(Alguns) Riscos associados a firewalls

- A firewall é inútil quando o tráfego não desejado é gerado na rede interna (“raposa dentro da capoeira”).

- A firewall é inútil contra IP spoofing. A máquina atacante assume o IP da máquina verdadeira.

- A firewall é inútil quando o serviço não permitido é oferecido noutra porta (que não está bloqueada).

- A firewall oferece uma segurança falsa se não estiver correctamente configurada.

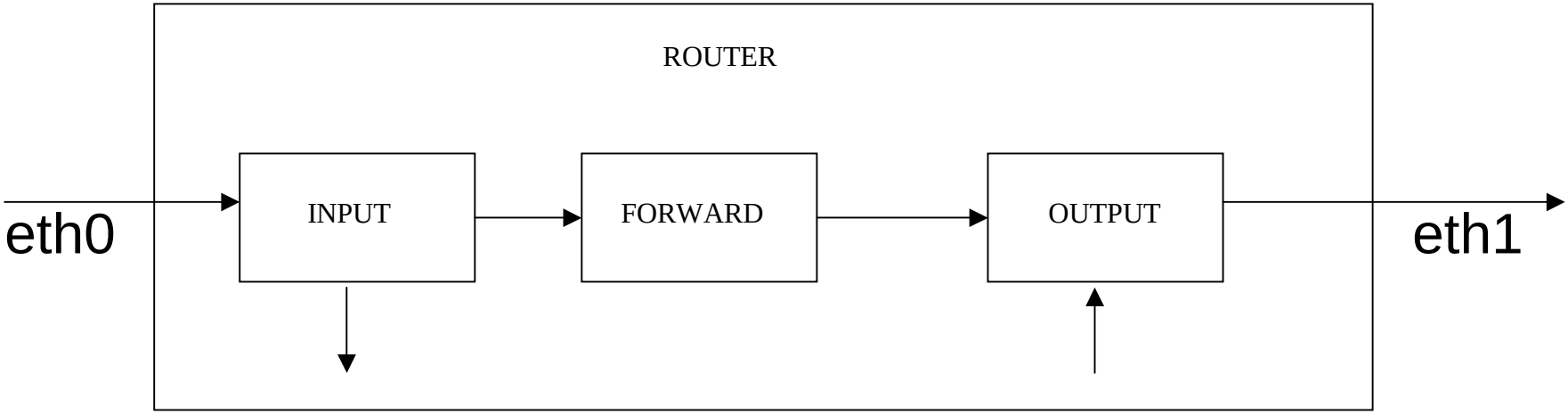
Linux IPchains

IPchains é um módulo do kernel linux que oferece regras de filtragem de pacotes de complexidade comparável a firewalls comerciais.

- tem regras que funcionam em todas as camadas (ligação, rede, transporte) já descritas;

- funciona com o conceito de cadeia de regras : as regras são analisadas sequencialmente até que o pacote obedeça a uma das regras.

- há três cadeias de regras standard: input chain, output chain, forward chain. Podem ser ainda geradas novas cadeias de regras (user defined chains).



- quando o pacote obedece a uma regra é executada uma acção (target action) : Deny, Reject, Accept, Masq, Redirect, Return,

IPchains sintaxe

ipchains comando regras opções

comandos

-P chain policy Define a acção de Default se o pacote não obedecer a nenhuma regra: ACCEPT, DENY, REJECT, REDIR

-A chain adiciona uma regra no fim da cadeira

-I chain adiciona uma regra no início da cadeira

-D chain apaga uma regra na cadeia

-N chain gera uma nova cadeia de regras (user defined chain)

-L faz uma listagem das regras em vigor

REGRAS

-p [!] protocolo TCP UDP ICMP all

-s [!] address/mask [!] porta

-d [!] address/mask [!] porta

-i [!] interface

-j target action ACCEPT, DENY, REJECT, REDIR, RETURN

OPÇÕES

-b aplica a regra nos dois sentidos

-n mostra IPs e portas (e não faz DNS lookups)

-y regra com TCP flags SYN bit set, ACK clear, FIN clear

exemplo

```
# ipchains   -P  forward  DENY
# ipchains   -A  forward  -s 192.168.1.0/24    -d 193.168.224.8/32 80 –p tcp -b  -j
DENY
# ipchains   -A  forward  -s 192.168.1.0/24    -d 0/0 80 –p tcp -b  -j ACCEPT
```

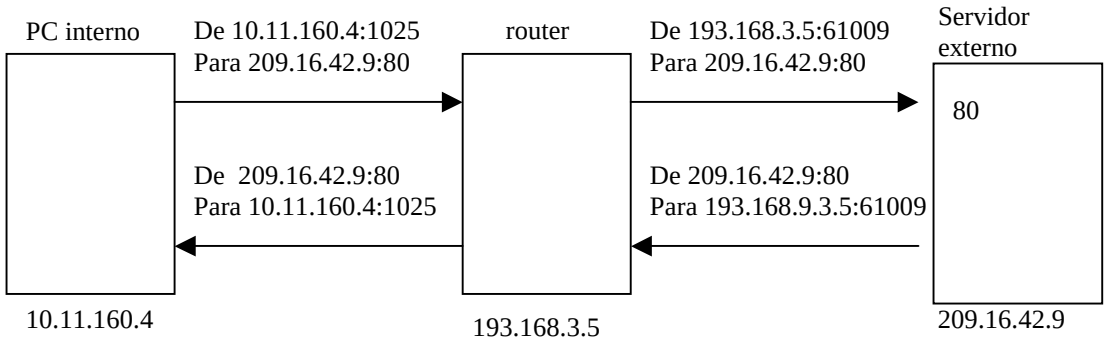
Network Address Translation (NAT)

Network address translation é uma função de “proxy” realizada ao nível da camada IP.

Essencialmente, um PC numa rede interna tem um IP privado que não é válido numa rede global.

Quando o PC na rede privada pretende realizar uma ligação para o exterior, o router que está de permeio faz a ligação em vez do PC interno e retorna a resposta ao PC interno.

Exemplo: NAT



Router: Tabela de ligações NAT

Origem	Destino	Porta aberta no router
10.11.160.4:1025	**209.16.42.9:80**	**61009**
10.11.153.34:4045	76.126.252.199:80	32568
10.11.22.173:35678	193.136.227.163:22	20567

- Comando ipchains

```
# ipchains -A  forward -s 10.11.0.0/16    -d 0/0 80 –p tcp   -j MASQ
```

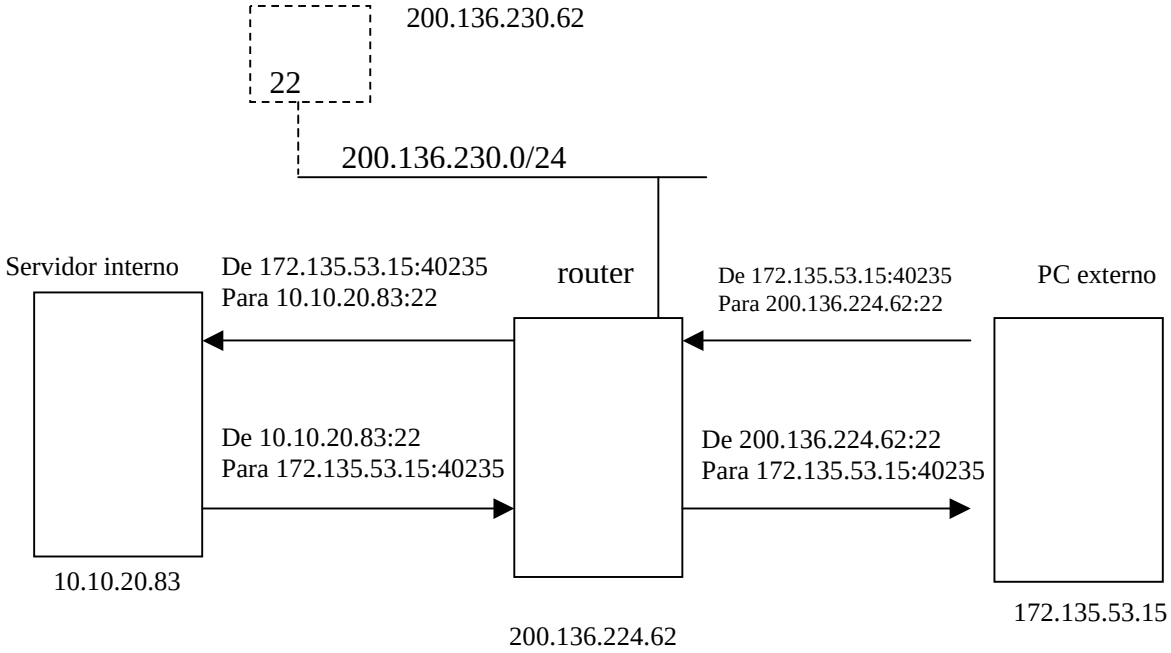
PORT FORWARDING. REVERSE NAT (RNAT)

- Essencialmente, permite dar acesso a partir da Internet a um servidor numa rede privada.

- O PC na Internet realiza uma ligação a um IP válido, o IP do router ou um IP de uma rede que o router anuncia na Internet.

- O router redirecciona o pacote IP para o PC interno e retorna a resposta do PC interno.

Exemplo: PORT FORWARDING



- comando Linux ipmasqadm

```
# ipmasqadm  portfw   -A   -p tcp   -L 192.136.224.62 22   -R 10.10.20.83 22
```

Exemplo: REVERSE NAT