Comando básicos iptables (RESUMO)

tabelas [tables] - é o nome de um conjunto de cadeias.

cadeias [chains] - é uma coleção de regras.

regras [rules] - é a condição usada para combinar o pacote.

alvo [target] - é a ação[1] realizada quando uma possível regra corresponde.

Exemplos do alvo são **ACCEPT**, **DROP**, **QUEUE, REJECT**.

Sintaxe:

iptables --table TABLE -A/-C/-D... CHAIN rule --jump Target

TABELA [TABLE]

Existem cinco tabelas possíveis:

FILTER

Tabela padrão usada para filtragem de pacotes. Inclui as seguintes cadeias:

- INPUT
- OUTPUT
- FORWARD

NAT

Tabela relacionada à tradução de endereços da rede. Inclui as seguintes cadeias:

- PREROUTING
- POSTROUTING
- OUTPUT

MANGLE

Tabela específica para ações que devem ser aplicadas no tráfego que passa pelas cadeias (dependendo das ações que passam pelo filter e nat.), essa tabela possui as seguintes cadeias:

- INPUT

[1] A política é a ação padrão tomada em caso de não correspondência com as cadeias embutidas e pode ser ACCEPT ou DROP.

- OUTPUT
- FORWARD
- PREROUTING
- POSTROUTING

RAW

Tabela que configura isenções de rastreamento de conexão. possui as seguintes cadeias:

- OUTPUT
- PREROUTING

CADEIAS [CHAINS]

Existem algumas cadeias integradas que são incluídas nas tabelas.

INPUT - Conjunto de regras para pacotes destinados a sockets localhost.
FORWARD - Para pacotes roteados por meio do dispositivo.
OUTPUT - Para pacotes gerados localmente, destinados a serem transmitidos externamente.
PREROUTING - Para alterar pacotes à medida que chegam ao firewall.
POSTROUTING - Para alterar os pacotes à medida que saem do firewall.

As cadeias definidas pelo usuário também podem ser criadas. Algumas opções que podem ser definidas são:

-A, –append: Acrescenta à cadeia fornecida nos parâmetros.

Sintaxe:

```
iptables [-t table] –append [chains] [parameters]
```

Exemplo: Este comando descarta todo tráfego vindo de qualquer porta.

```
iptables -t filter –append INPUT -j DROP
```

-D, –delete: Exclui a regra da cadeia especificada.

Sintaxe:

```
iptables [-t table] –delete [chain] [rule_number]
```

Exemplo: Este comando irá excluir a regra 2 da cadeia INPUT.

```
iptables -t filter –delete INPUT 2
```

-C, –check: Analise se uma regra está presente na cadeia ou não. Ele retorna 0 se a regra existir, caso contrário 1.

Sintaxe:

```
iptables [-t table] –check [chains] [parameters]
```

Exemplo: Este comando verifica se a regra está especificada está presente na cadeia INPUT.

```
iptables -t filter –check INPUT -s 192.168.0.1 -j DROP
```

PARÂMETROS

Os parâmetros fornecidos com o comando iptables são usados para combinar o pacote e executar a ação especificada. Os parâmetros comuns são:

-p, –proto: é o protocolo que o pacote segue. Os valores possíveis podem ser tcp, udp, icmp, ssh etc.

Sintaxe:

```
iptables [-t table] –A [chains] -p {protocol_name} <target>
```

Exemplo 1.1 este comando irá anexar uma regra na cadeia INPUT para descartar todos os pacotes udp.

```
iptables -t filter –A INPUT -p udp -j DROP
```

Exemplo 1.1

-s, –source: é usado para combinar com o endereço de origem do pacote.

Sintaxe:

```
iptables [-t table] –A [chains] -s {source_address} <target>
```

Exemplo 1.2 este comando irá anexar uma regra na cadeia INPUT para aceitar todos os pacotes de origem do endereço 192.168.0.12.

```
iptables -t filter -A INPUT -s 192.168.1.230 -j ACCEPT
```
exemplo 1.2

-d, –destination: é usado para combinar com o endereço de destino do pacote.

Sintaxe:
```
iptables [-t table] -A [chain] -d {destination_address} <target>
```

Exemplo 1.3 este comando anexa uma regra na cadeia OUTPUT para descartar todos os pacotes destinados a 192.168.1.66.
```
iptables -t filter -A OUTPUT -d 192.168.1.66  -j DROP
```
exemplo 1.3

-i, –in-interface: combina pacotes com a interface especificada e executa a ação.

Sintaxe:
```
iptables [-t table] -A [chain] -i {interface} <target>
```

Exemplo 1.4: Este comando incrementa uma regra na cadeia INPUT para descartar todos os pacotes destinados à interface[2] sem fio.
```
iptables -t filter -A INPUT -i  wlan0 -j DROP
```
exemplo 1.4

-o, –out-interface: combina os pacotes com a interface de saída especificada.
-j, –jump: este parâmetro especifica a ação a ser executada em uma partida.

Sintaxe:
```
iptables [-t table] -A [cadeia] [parametro] -j <destino>
```

Exemplo 1.5: Este comando adiciona uma regra na cadeia FORWARD para descartar todos os pacotes.
```
iptables -t filter -A FORWARD -j  DROP
```
exemplo 1.5

[2] Ou interface de rede. pode examinar a interface da rede, utilizando o comando ‘netstat -i’ ou ‘ip link show’

PROTEÇÃO E SEGURANÇA.

Por mais que seu sistema seja seguro ou segue algumas medidas de segurança, é necessário ainda propor medidas implementadas em seu sistema, para dificultar a vida de possíveis invasores ao seu sistema. Apesar do sistema seguir regras e alguns protocolos de segurança, muitas das vezes crackers procuram brechas no sistemas para realizar ataques ou descobrir possíveis falhas que possam ser exploradas. Para isso muitos desses **black hats**[3] utilizam scanner ou aplicações que podem procurar brechas a serem estudadas para um possível ataque. Para isso, uma das medidas que podem dificultar esse escaneamento da rede ou bloqueio desses pacotes é usar regras específicas para isso. Abaixo segue alguns comandos muito interessantes.

O comando irá bloquear a entrada de pacotes icmp na sua máquina. Se um invasor tentar enviar um ping para a sua máquina, esse pacote não irá entrar na sua máquina. Exibindo uma mensagem para o mesmo, dizendo que houve perda 100% dos pacotes enviados para o destino.

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

Bloquear porta tcp

```
iptables -I INPUT -p tcp --dport 22 -j DROP
```

Bloquear porta para um destino específico

```
iptables -I INPUT -p tcp -s 10.0.0.0/8 --dport 22 -j DROP
```

Bloquear porta e logar tentativas de conexão

```
iptables -I INPUT -p tcp --dport 22 -j LOG
iptables -A INPUT -p tcp --dport 22 -j DROP
```

Bloqueia mais de uma porta ao mesmo tempo

```
iptables -I INPUT -p tcp -m multiport --dports 1,8080 -j DROP
```

bloqueia ou aceita para um interface de rede especificada.

```
iptables -I INPUT -i eth0 -p tcp --dport 80 -j ACCEPT/DROP
```

ALGUMAS OBSERVAÇÕES IMPORTANTES A SALIENTAR.

[3] Black hats é um termo usado pela comunidade hacker para definir uma categoria de hackers que usam seus conhecimentos de invasão para o mal. Antigamente muitas pessoas leigas, definiam o termo hacker com a conotação negativa. Atualmente devido ao crescimento da internet o termo é utilizado para definir cibercriminosos ou hackers do mal.

Antes de utilizar o iptables, certifique que utilizando juntamente com o comando sudo ou que esteja configurado o acesso ao terminal no modo root

exemplo:

```
mvictor@mvictor-OEM:~$ sudo iptables –list
```

ou

```
mvictor@mvictor-OEM:~$ sudo su && iptables –list
```

Se você já estiver com acesso a super usuário no linux, não tem a necessidade de usar o comando **sudo,** podendo executar o comando diretamente do terminal.

```
root@mvictor-OEM:/home/mvictor# iptables –list
```

Outro opção bastante interessante é criar um script shell que configure opções automaticamente, dispensando a necessidade do usuário digitar manualmente uma quantidade de comandos que podem ser repetidos para um número de alvos específicos. Pensem em um contexto em que você tenha uma lista de endereços ips que precisam ser bloqueados na rede, nesse caso um único comando poderá ser utilizado, porém a empresa tem a necessidade de bloquear uma lista de endereços ips, de terem acesso ao seu servidor, ou algum serviço que ele provém. Simplesmente o que você faz é, criar uma lista com esses ips, em seguida criar um script automatiza essa tarefa, em seguida você adiciona esses comandos gerados com esses ips dessa lista, e crie um script shell para executar essas regras do iptables. Muito bacana essa ideia né?!!

Ao utilizar os comandos, você pode remover todas as regras de filtragem e cadeias criadas pelo usuário.

```
sudo iptables –flush
```

Para salvar as configurações do iptable, digite o seguinte comando no terminal:

```
sudo iptables-save
```

A restauração das configurações do iptables pode ser feita por meio do comando:

```
sudo iptables-restore
```

Existem outras interfaces, como ip6tables, que são usadas para gerenciar tabelas de filtragem para IPv6.

REFERÊNCIAS BIOGRÁFICAS

NETO, Urubatan, Dominando Linux Firewall Iptables. Rio de Janeiro: Editora Ciência Moderna, 2004.

Cisco, O que é um firewall, CIsco, 2022. Disponível em: <https://www.cisco.com/c/pt_br/products/security/firewalls/what-is–firewall.html#~related-topics>. Acesso em 04/04/2022.

CASTRO, Lucas, Introdução ao iptables. Iron Linux, 2020. Disponível em <https://ironlinux.com.br/introducao-ao-iptables/>. Acesso em: 02/04/2022.

W!ME, Como configurar um firewall com iptables no ubuntu 14.04. Iron linux, W!me, 2015. Disponível em: <https://wime.com.br/2015/03/04/como-configurar-um-firewall-usando-o-iptables-ubuntu-14-04/>. Acesso em: 02/04/2022.