Universidade Federal de Lavras
Departamento de Ciência da Computação

Segurança com Iptables

Alunos : Felipe Gutierrez e Ronan de Brito Mendes

Lavras – MG
11/2008

Sumário

1 - Introdução

Nos sabemos que nos dias de hoje é essencial o firewall, em servidores corporativos como servidores de web, e-mail e gateways, devido a demanda de hackers e lammers. Mas sabemos que assim mesmo não basta montar um firewall e por em mente que nunca vai ser invadido, lembra-se de que é primordial verificar Bugs e falhas de sistema. Tenha em mente além de configurar um sistema de firewall, também administratar por Logs e IDS, e que tenha administrar os logs para verificar possíveis erros e de uma forma fácil e ágil, e IDS para verificar a Detecção de Intrusos.

Quando falamos em sistemas operacional linux como firewall você encontra ferramentas capazes e eficazes para efetuar esta função. E o melhor de tudo como que o Linux é free, você pode adquirir qualquer distribuição Linux , pedindo em sites por exemplo , e instalar em seu pc sem nenhum problema. Recomendamos a distribuição Ubuntu , por ser de fácil instalação e fácil manuseio.

2 – Softwares de Firewall para o Linux

Conheça os Firewalls conforme a versão do kernel :

- **Ipfwadm** - O IP Firewall Administration, ou simplesmente ipfwadmin foi a ferramenta padrão para construção de regras de firewall, para o Kernel anterior à versão 2.2.0. Dizem que o Ipfwadm era extremamente complexo.
- **Ipchains** - O ipchains foi a solução, ou melhor, a atualização, feita para o kernel 2.2 do ipfwadm. A idéia do ipchains foi ter o poder do ipfwadm, mas com uma simplicidade e facilidade no que diz respeito à criação de regras. Além de prover sua facilidades, é criar uma compatibilidade com o ipfwadm através do utilitário ipfwadm-wrapper.
- **Iptables** e o **Netfilter** - A nova geração de ferramentas de firewall para o Kernel 2.4 do Linux. Além de possuir as facilidade do ipchains, e implementar a facilidade do NAT e filtragem de pacotes mais flexíveis que o IPchains. Para saber mais informações do Iptables acessem http://www.netfilter.org/ .

3 – Regras do Firewall

Na configuração do Firewall com o iptables, é preciso saber quais são as regras a serem utilizadas para rodar o Firewall.

Regras do Firewall :

- **INPUT :** È utilizada quando o destino final é a própria máquina firewall.
- **OUTPUT :** Qualquer pacote gerado pela máquina firewall e que deva sair para a rede será tratado pela regra OUTPUT.
- **FORWARD :** Qualquer pacote que atravessa o firewall, de uma máquina e direcionado à outra, será tratado pela chain FORWARD.

Basicamente o IPTABLES tem as seguintes políticas:

- **DROP :** Nega pacote e não manda um pacote de volta para o emitente.
- **ACCEPT :** Aceita o pacote
- **REJECT :** Nega pacote e manda um pacote de volta do tipo host-unreachable (Host Inalcançável)

4 - Comandos Principais do IPtables

- **-A :** Este comando acrescenta uma regra às existentes no sistema, ou seja, permite atualizar regras já existentes na estrutura do firewall.
- **-I :** Este comando insere uma nova regra dentro das existentes no firewall.
- **-D :** Este comando exclui uma regra específica no firewall.
- **-P :** Este comando define a regra padrão do firewall.
- **-L :** Este comando lista as regras existentes no firewall.
- **-F :** Este comando ZERA todas as regras criadas no Firewall (o chamado flush).
- **-h :** Este comando mostrará o help, ajuda de comando.
- **-R :** Este comando substitui um regra no firewall.
- **-C :** Este comando basicamente checa as regras.
- **-Z :** Este comando zera uma regra específica.
- **-N :** Este comando cria uma nova regra com um nome.
- **-X :** Este comando exclui uma regra específica por seu nome.

5 – Parâmetros

Os parâmetros padrão do iptables são os seguintes:

- **-p!** (protocolo): define qual o protocolo TCP/IP deverá ser tratado. São

eles: TCP, UDP e ICMP

- **-s!** (origem)/ **-d!** (destino): Define qual o endereço de origem (-S) e de destino (-D) que a regra atuará. Este comando possui dois argumentos: endereço/máscara e porta.

  -i! (interface): define o nome da interface de rede onde trafegará os

  pacotes de entrada e saída do firewall. Muito utilizado em mascaramento e técnicas de NAT. Exemplo: -W eth1.

- **-j!** (ir para): Serve para redirecionar uma ação desde que as regras sejam

similares.

- **-f!** (fragmento): Trata datagrama fragmentados.

6 - Extensões

Novidade do iptables que facilita as regras.

- **sport[!] [port:port] -dport[!] [port:port]** : Normalmente estas extensões são utilizadas com o comando -m do iptables. Trata-se de um direcionamento de porta(s) origem (-sport), para porta(s) destino (-dport). Pode-se inclusive definir um número padrão de portas para o acesso (port:port). Este comando pode ser utilizado tanto para portas TCP ou UDP.
- **mac-source[!] endereço** : especifica qual a placa de rede, através de seu endereço MAC, que irá transmitir pacotes através do firewall, limitado pela

política do mesmo.

- **icmp-type[1] tipo** : especifica quais os tipos de pacotes ICMP pode passar ou não pelo firewall, São eles:

- **Mensagem Tipo Código**

```
Echo-request 8 0
Echo-reply 3 0
Source-quench 4 0
Time-exceed 11 0
Destination-unreachable 3 0
Network-unreachable 3 0
Host-unreachable 3 1
Protocol-unreachable 3 2
Port-unreachable 3 3
```

Com isto podemos bloquear alguns ataques do tipo ping flood, bloquear ping e etc

- **[!] -- syn** : especifica o uso dos bits ACK e FIN em requisições SYN TCP.

Especificamente, a opção `-m state' aceita uma opção adicional `--state', que é uma lista de estados de ativação separados por vírgula. (a flag '!' não indica a ativação desses estados). Esses estados são:

- **NEW** : um pacote que cria uma nova conexão.

- **ESTABLISHED** : um pacote que pertence a uma conexão existente (isto é, um pacote de resposta).

- **RELATED** : um pacote que está relacionado com (mas não faz parte de) uma conexão existente, como um ICMP error, ou (com o módulo FTP inserido),um pacote que é estabelecido por uma conexão de dados ftp.

– **INVALID** : Um pacote que não poderia ser identificado por alguma razão: isto inclui execução fora da memória e erros de ICMP que não correspondam a nenhuma conexão existente. Geralmente estes pacotes devem ser barrados (drop).

7 – Exemplo do Firewall

```
# iptables -A INPUT -p icmp -j DROP
Esta regra nega todos os pacotes ICMP vindos do servidor, em que se encontra o
firewall.

# iptables -D INPUT -p icmp -j DROP
Esta regra exclui a regra criar acima.

# iptables -A INPUT -s 192.168.1.0/24 -j DROP
Esta regra acima faz com que todos os pacotes vindo de qualquer endereço da
classe de ip 192.168.1.1 á 192.168.1.255 nega os pacotes.

# iptables -A OUTPUT -p icmp -d! 192.168.1.0/24 -j ACCEPT
Esta regra acima faz com que todos os pacotes vindo de qualquer endereço da
classe de ip 192.168.1.1 á 192.168.1.255 aceita os pacotes.

#echo 1 > /proc/sys/net/ipv4/ip_forward
Habilitando o recurso de IP forwarding
```

8 – Configurando o Firewall contra ataque

Proteção contra Syn-floods

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

Port scanners ocultos

```
# iptables -A FORWARD -p tcp --tcp-flags SYN,ACK,FIN,RST RST -m limit --limit
1/s -j ACCEPT
```

Ping da morte

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

Proteção Contra IP Spoofing

```
# iptables -A INPUT -s 10.0.0.0/8 -i Interface da NET -j DROP
# iptables -A INPUT -s 172.16.0.0/16 -i Interface da NET -j DROP
# iptables -A INPUT -s 192.168.0.0/24 -i Interface da NET -j DROP
```

<u>Obs.: Interface da NET pode ser ppp0, ethX e etc.</u>

Log a portas proibidas e alguns backdoors

Porta FTP

```
# iptables -A INPUT -p tcp --dport 21 -j LOG --log-prefix "Serviço: FTP"

Porta Wincrash
# iptables -A INPUT -p tcp --dport 5042 -j LOG --log-prefix "Serviço: Wincrash"

Portas BackOrifice
# iptables -A INPUT -p tcp --dport 12345 -j LOG --log-prefix "Serviço:
BackOrifice"
# iptables -A INPUT -p tcp --dport 123456 -j LOG --log-prefix "Serviço:
BackOrifice"
```