Capítulo 7 - Servidor Firewall e Aplicações

Grupo ASI - IFG Câmpus Formosa

1 Ficha Técnica

Serviço	**Descrição**
Método de Comunicação	Protocolo TCP/UDP, portas configuráveis (22, 80, 443, etc.)
Funções	Controle de tráfego de rede, filtragem de pacotes, NAT, balanceamento de carga
Pacote para instalação	iptables (já incluído no kernel Linux), iptables-persistent
Script de controle	/etc/init.d/netfilter-persistent, systemctl status iptables

Tabela 1: Ficha Técnica do Servidor iptables

2 Descrição do Servidor

O iptables é um firewall baseado em kernel do Linux que permite controlar o tráfego de rede através de regras de filtragem de pacotes. É uma ferramenta fundamental para segurança de rede, permitindo:

- **Filtragem de Pacotes**: Controlar quais pacotes podem entrar ou sair do sistema
- **NAT (Network Address Translation)**: Tradução de endereços de rede
- **Mangle**: Modificação de cabeçalhos de pacotes
- **Logging**: Registro de atividades de rede para auditoria

O iptables trabalha com três tabelas principais:

- **filter**: Tabela padrão para filtragem de pacotes
- **nat**: Para tradução de endereços de rede
- **mangle**: Para modificação de pacotes

E cinco cadeias (chains) principais:

- **INPUT**: Pacotes destinados ao sistema local
- **OUTPUT**: Pacotes originados do sistema local
- **FORWARD**: Pacotes que passam pelo sistema (roteamento)
- **PREROUTING**: Pacotes que chegam (NAT)
- **POSTROUTING**: Pacotes que saem (NAT)

3 Instalação

O iptables já vem incluído no kernel Linux, mas para uma configuração completa, é necessário instalar alguns pacotes adicionais:

Listing 1: Instalação do iptables

```
# Atualizar repositórios
sudo apt update

# Instalar iptables e ferramentas relacionadas
sudo apt install iptables iptables-persistent netfilter-
   persistent

# Verificar se o iptables está funcionando
sudo iptables -L -v
```

Para verificar se o módulo do kernel está carregado:

Listing 2: Verificação de módulos

```
# Verificar módulos carregados
lsmod | grep iptable

# Carregar módulos se necessário
sudo modprobe iptable_filter
sudo modprobe iptable_nat
sudo modprobe iptable_mangle
```

4 Arquivos de Configuração e Principais Características

4.1 Arquivos de Configuração

- **/etc/iptables/rules.v4**: Arquivo principal de regras IPv4
- **/etc/iptables/rules.v6**: Arquivo principal de regras IPv6
- **/etc/default/iptables**: Configurações padrão
- **/proc/net/ip_tables_names**: Tabelas ativas

4.2 Comandos Principais

Listing 3: Comandos básicos do iptables

```
# Listar todas as regras
sudo iptables -L -v -n

# Listar regras com números de linha
sudo iptables -L -v -n --line-numbers

# Limpar todas as regras
sudo iptables -F

# Definir políticas padrão
sudo iptables -P INPUT DROP
sudo iptables -P OUTPUT ACCEPT
sudo iptables -P FORWARD DROP
```

4.3 Exemplo de Configuração Básica

Listing 4: Configuração básica de firewall

```
#!/bin/bash

# Limpar todas as regras
iptables -F
iptables -X
iptables -t nat -F
iptables -t nat -X
iptables -t mangle -F
```

```
iptables -t mangle -X

# Definir políticas padrão
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Permitir tráfego local
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Permitir conexões estabelecidas
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Permitir SSH (porta 22)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Permitir HTTP (porta 80)
iptables -A INPUT -p tcp --dport 80 -j ACCEPT

# Permitir HTTPS (porta 443)
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# Permitir ping (ICMP)
iptables -A INPUT -p icmp -j ACCEPT
```

5 Outras opções/Descritivas de segurança/Boas práticas/Exemplos

5.1 Configuração de NAT

Listing 5: Configuração de NAT

```
# Habilitar NAT para rede interna
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

# Redirecionar porta externa para interna
iptables -t nat -A PREROUTING -p tcp --dport 8080 -j DNAT --to-
   dest 192.168.1.100:80
```

5.2 Configuração de Logging

Listing 6: Configuração de logs

```
# Log de tentativas de conexão SSH
iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "
    SSH_ATTEMPT: "

# Log de pacotes rejeitados
iptables -A INPUT -j LOG --log-prefix "DROP: "
```

5.3 Proteção contra Ataques

Listing 7: Proteções de segurança

```
# Proteção contra SYN flood
iptables -A INPUT -p tcp --syn -m limit --limit 1/s --limit-burst
    3 -j ACCEPT

# Proteção contra port scanning
iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP

# Proteção contra ataques de força bruta SSH
iptables -A INPUT -p tcp --dport 22 -m recent --name SSH --set
iptables -A INPUT -p tcp --dport 22 -m recent --name SSH --update
    --seconds 60 --hitcount 4 -j DROP
```

5.4 Boas Práticas

1. **Sempre testar regras em ambiente de desenvolvimento**
2. **Manter backup das configurações atuais**
3. **Documentar todas as regras criadas**
4. **Monitorar logs regularmente**
5. **Usar políticas restritivas por padrão**
6. **Implementar rate limiting para serviços críticos**
7. **Manter o sistema atualizado**

5.5 Script de Backup e Restauração

Listing 8: Script de backup

```
#!/bin/bash

# Backup das regras atuais
iptables-save > /backup/iptables_rules_$(date +%Y%m%d_%H%M%S).bak

# Restaurar regras
# iptables-restore < /backup/iptables_rules_20231201_143022.bak
```

5.6 Monitoramento e Manutenção

Listing 9: Comandos de monitoramento

```
# Verificar estatísticas
sudo iptables -L -v -n

# Monitorar logs em tempo real
sudo tail -f /var/log/syslog | grep iptables

# Verificar conexões ativas
sudo netstat -tuln

# Verificar processos de rede
sudo ss -tuln
```

6 Considerações finais

O iptables é uma ferramenta poderosa e essencial para a segurança de redes Linux. Sua flexibilidade permite implementar políticas de segurança complexas, desde configurações básicas até setups avançados de alta disponibilidade.

Principais pontos a considerar:

- **Complexidade**: A curva de aprendizado pode ser íngreme, mas o domínio da ferramenta é fundamental
- **Performance**: Regras mal configuradas podem impactar o desempenho da rede
- **Manutenção**: Configurações devem ser revisadas e atualizadas regularmente

- **Documentação**: Manter documentação atualizada é crucial para troubleshooting
- **Testes**: Sempre testar configurações em ambiente controlado antes da produção

Para ambientes de produção, considere também:

- Implementar failover para alta disponibilidade
- Usar ferramentas de monitoramento como Nagios ou Zabbix
- Implementar alertas automáticos para tentativas de intrusão
- Manter procedimentos de recuperação de desastres
- Treinar equipe de suporte nas configurações implementadas