¿Qué es IPTABLES?

IPTABLES es una herramienta de firewall en sistemas Linux que permite filtrar, modificar y redirigir el tráfico de red. Funciona mediante reglas que definen cómo se manejan los paquetes de red (aceptar, rechazar o descartar).

Conceptos Básicos de IPTABLES

1. **Tablas**: IPTABLES organiza las reglas en tablas. Las tablas más comunes son:
    - **filter**: Para filtrar paquetes (aceptar, rechazar o descartar).
    - **nat**: Para traducción de direcciones de red (NAT).
    - **mangle**: Para modificar paquetes (por ejemplo, cambiar el TTL).
2. **Cadenas**: Cada tabla tiene cadenas predefinidas que representan puntos de decisión:
    - **INPUT**: Paquetes destinados al sistema local.
    - **OUTPUT**: Paquetes generados por el sistema local.
    - **FORWARD**: Paquetes que se enrutarán a través del sistema.
    - **PREROUTING** y **POSTROUTING**: Para NAT y mangle.
3. **Reglas**: Son las instrucciones que definen qué hacer con los paquetes. Cada regla tiene:
    - Un criterio de coincidencia (por ejemplo, dirección IP, puerto, protocolo).
    - Una acción (ACCEPT, DROP, REJECT).
4. **Argumentos**:
    1. `-A <CHAIN>` : Añade regla a la cadena especificada
    2. `-D <CHAIN>` : Eliminar Regla
    3. `-s <SOURCE>` : Fuente, IP de procedencia
    4. `-j <ACTION>` : (jump) - Salto (ACCEPT, DROP or REJECT)
    5. `-p <protocol>` : Protocolo
    6. `--dport <port>`: Destination Port
    7. `--sport: <port>`: Puerto fuente,
    8. `-i` :interfaz entrante
    9. `-o` :interfaz saliente
    10. `-s` :dirección IP origen
    11. `-d` :dirección IP destino

Diseño de un Firewall con IPTABLES

1. **Definir políticas por defecto**: Establece qué hacer con el tráfico que no coincide con ninguna regla.
    - Ejemplo: `iptables -P INPUT DROP` (rechazar todo el tráfico entrante por defecto). (-P: policy, --p: policy)
2. **Permitir tráfico necesario**: Abre solo los puertos y servicios esenciales.

- Ejemplo: `iptables -A INPUT -p tcp --dport 22 -j ACCEPT` (permitir SSH). (-A:Add, -p: protocol, --dport: Destination Port, -j: JUMP)

3. **Bloquear tráfico no deseado**: Bloquea direcciones IP o rangos sospechosos.
   - Ejemplo: `iptables -A INPUT -s 192.168.1.100 -j DROP`.
4. **Habilitar NAT (si es necesario)**: Para permitir que los dispositivos internos accedan a Internet.
   - Ejemplo: `iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE`.

Configuración Básica de IPTABLES

1. **Listar reglas existentes**:

```
iptables -L -v -n
```

2. **Agregar una regla**: Permitir tráfico HTTP (puerto 80):

```
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

1. **Eliminar una regla**: Eliminar la regla de HTTP:

```
iptables -D INPUT -p tcp --dport 80 -j ACCEPT
```

1. **Guardar reglas**:

- En sistemas basados en Debian/Ubuntu:

```
iptables-save > /etc/iptables/rules.v4
```

- En sistemas basados en RedHat/CentOS:

```
service iptables save
```

Administración y Gestión de IPTABLES

1. **Monitoreo del tráfico**:

- Usa `iptables -L -v -n` para ver el tráfico que coincide con las reglas.

2. **Bloquear ataques**:

- Limitar el número de conexiones por IP para evitar ataques DDoS:

```
iptables -A INPUT -p tcp --dport 80 -m limit --limit 25/minute -j
ACCEPT
```

3. **Reglas persistentes**:

- Asegúrate de que las reglas se guarden y se carguen al reiniciar el sistema.

4. **Logging**:

- Registrar tráfico sospechoso:

```
iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "SSH Attempt:
"
```

Ejemplo de Configuración Básica

```
# Establecer políticas por defecto
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Permitir tráfico local
iptables -A INPUT -i lo -j ACCEPT

# Permitir tráfico HTTP y HTTPS
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# Permitir SSH desde una IP específica
iptables -A INPUT -p tcp --dport 22 -s 192.168.1.50 -j ACCEPT

# Habilitar NAT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

# Guardar reglas
iptables-save > /etc/iptables/rules.v4
```

Consejos de Seguridad

1. Minimizar la exposición: Abre solo los puertos necesarios.
2. Actualizar regularmente: Mantén tu sistema y aplicaciones actualizados.
3. Monitorear registros: Revisa los logs para detectar actividades sospechosas.
4. Usar herramientas adicionales: Combina IPTABLES con herramientas como Fail2Ban para mayor seguridad.