Conteúdos

1. Serviços de rede

Nesta secção, vamos explorar como os serviços de rede são geridos no Linux, desde o seu início até ao encerramento, e os principais ficheiros e diretórios envolvidos neste processo.

- **Gestão de Serviços com `systemd`**: O `systemd` é o sistema de inicialização e gestor de serviços padrão na maioria das distribuições Linux modernas. Ele usa unidades para gerir os serviços de forma mais robusta e eficiente. Os comandos mais comuns são:
  - `systemctl start [serviço]`: Inicia um serviço.
  - `systemctl stop [serviço]`: Para um serviço.
  - `systemctl status [serviço]`: Verifica o estado de um serviço.
  - `systemctl enable [serviço]`: Habilita um serviço para iniciar automaticamente no boot.
- **Ficheiro `/etc/services`**: Uma base de dados de mapeamento entre nomes de serviços e números de porta. É utilizada pelo sistema e pelas aplicações para identificar serviços de rede de forma legível.
- **Lista de portas e serviços**: Usar o comando `ss` (socket statistics) é uma forma moderna e rápida de inspecionar as conexões e portas abertas.

Exemplo Prático: Gerir o Serviço SSH Vamos verificar o estado do serviço SSH. O SSH (Secure Shell) é um serviço que permite a conexão remota segura a um servidor.

```
$ systemctl status sshd
```

A resposta típica mostra o estado do serviço (ativo ou inativo), o PID (Process ID) e a quantidade de tempo que está a correr.

Exercício Prático: Gerir o Serviço HTTP 1. Verifique o estado do serviço de servidor web Apache (`httpd`). 2. Tente pará-lo usando o comando apropriado. 3. Verifique novamente o estado para confirmar que foi parado.

2. XINET.d

O `xinetd` (e o seu antecessor, o `inetd`) funciona como um "super-servidor"que gere a inicialização de serviços de rede que não precisam de estar ativos a todo o momento. Ele espera por pedidos de conexão numa porta específica e, quando um pedido chega, inicia o serviço correspondente.

- **Arquivos de Configuração**:
  - `/etc/xinetd.conf`: O ficheiro de configuração principal, que define o seu comportamento global.
  - `/etc/xinet.d/`: Este diretório contém ficheiros de configuração individuais para cada serviço gerido pelo `xinetd`.

Exemplo Prático: Configuração de um Serviço Um ficheiro de configuração em `/etc/xinet.d/` para o serviço `telnet` pode ter a seguinte aparência. Note a linha `disable = yes`, que evita que o serviço inicie.

```
service telnet
{
disable = yes
id = telnet-ipv4
type = UNLISTED
...
}
```

Exercício Prático: Habilitar um Serviço 1. Encontre um serviço no diretório `/etc/xinet.d/` que esteja desabilitado. 2. Altere o ficheiro de configuração desse serviço para habilitá-lo, alterando o valor `disable = yes`. 3. Reinicie o serviço `xinetd` para aplicar a alteração.

3. TCPWrappers

Uma ferramenta de segurança simples mas eficaz para controlar o acesso a serviços de rede. O `TCPWrappers` permite a criação de regras de acesso (permitir/negar) baseadas em endereços IP, nomes de host e nomes de utilizador.

- `/etc/hosts.allow`: Ficheiro que define as regras de "permitir".
- `/etc/hosts.deny`: Ficheiro que define as regras de "negar".

Exemplo Prático: Regras de Acesso Vamos supor que queremos permitir que o serviço SSH seja acedido apenas a partir de um IP específico e negar todo o restante tráfego.

```
Em /etc/hosts.allow:
sshd: 192.168.1.100

Em /etc/hosts.deny:
sshd: ALL
```

Nota: As regras no hosts.allow são processadas primeiro. Se uma regra corresponder, a conexão é permitida e o hosts.deny é ignorado.

Exercício Prático: Proteger o SSH 1. Adicione uma regra ao `/etc/hosts.deny` para bloquear o acesso de qualquer IP ao serviço SSH. 2. Teste a sua configuração a partir de outra máquina.

4. NIS

O NIS (Network Information Service) é um sistema de diretório centralizado que permite que informações de contas de utilizadores e hosts sejam distribuídas por uma rede. É útil para ambientes de rede pequenos e uniformes.

- **Configuração do Servidor e Cliente**: A configuração envolve a instalação dos pacotes necessários (`ypserv` para o servidor e `ypbind` para o cliente) e a definição de um domínio NIS.
- **Ficheiro `/etc/yp.conf`**: Ficheiro de configuração do cliente que especifica o domínio e o servidor NIS a ser utilizado.

Exemplo Prático: Listar Utilizadores NIS Após a configuração do cliente, podemos listar os utilizadores do servidor NIS com o comando `ypcat`.

```
$ ypcat passwd
```

Este comando mostra o conteúdo do mapa `passwd` do NIS, que é uma lista dos utilizadores e suas informações.

Exercício Prático: Verificar o Domínio 1. Use um comando para verificar o domínio NIS da sua máquina. 2. Verifique se o serviço `ypbind` está a correr.

5. DHCP

O DHCP (Dynamic Host Configuration Protocol) é o protocolo padrão para atribuir configurações de rede (como endereços IP) a dispositivos de forma automática.

- **Conceito**: A atribuição automática de IPs evita erros de configuração e torna a gestão da rede mais eficiente.
- **Configuração**: Toda a configuração de DHCP é feita no ficheiro `/etc/dhcp/dhcpd.conf`.
- **Parâmetros Chave**:
  - `range`: A gama de IPs disponíveis para atribuição.
  - `routers`: O endereço do gateway padrão da rede.
  - `domain-name-servers`: Os endereços dos servidores DNS para os clientes.

Exemplo Prático: Configurar um Sub-rede Um exemplo de uma configuração simples para a sub-rede 192.168.1.0/24:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.100 192.168.1.200;
option routers 192.168.1.1;
option domain-name-servers 8.8.8.8, 8.8.4.4;
default-lease-time 600;
max-lease-time 7200;
}
```

Exercício Prático: Configuração Estática 1. Crie uma entrada no seu ficheiro `dhcpd.conf` para atribuir um IP estático (por exemplo, 192.168.1.50) a uma máquina específica, usando o seu endereço MAC. 2. Após a alteração, reinicie o serviço DHCP.

6. DNS

O DNS (Domain Name System) é a base da internet, atuando como um "livro de endereços"que traduz nomes de domínio em endereços IP.

- **Conceitos Fundamentais**: Entender termos como Zona, Domínio, e Servidores Matriz (`root servers`) é crucial.
- **BIND (`named`)**: O software de servidor DNS mais utilizado. A sua configuração principal é no ficheiro `/etc/named.conf`.
- **Zonas**: As zonas são ficheiros de texto que contêm os registos para um domínio específico. Os tipos de registo mais comuns são:
  - **A**: Mapeia um nome de host para um endereço IPv4.
  - **MX**: Define o servidor de e-mail para o domínio.
  - **CNAME**: Cria um alias para um nome de host existente.

Exemplo Prático: Ficheiro de Zona Conteúdo de um ficheiro de zona simples (`db.exemplo.com`):

```
$TTL 86400
@ IN SOA ns1.exemplo.com. admin.exemplo.com. (
2023010101 ; Serial
3600       ; Refresh
1800       ; Retry
604800     ; Expire
86400      ; Minimum TTL
)

@   IN  NS  ns1.exemplo.com.
@   IN  A   192.168.1.10

www IN  A   192.168.1.11
mail IN A   192.168.1.12
```

Exercício Prático: Adicionar um Registo 1. No ficheiro de zona, adicione um novo registo `A` para um servidor de blog, com o nome `blog.exemplo.com` e o IP `192.168.1.20`. 2. Após a alteração, incremente o número de série (Serial) para que as alterações sejam propagadas. 3. Recarregue o serviço DNS para aplicar as alterações.

7. LOGS

Os logs são ficheiros de registo que fornecem informações sobre o que está a acontecer no sistema e nas aplicações. São cruciais para a monitorização e a resolução de problemas.

- **Pasta `/var/log`**: O diretório padrão onde a maioria dos logs do sistema e de aplicações é armazenada.

- **Ficheiro `messages`**: Contém mensagens gerais do sistema, do kernel e de serviços, sendo um dos primeiros lugares para procurar quando algo corre mal.
- **Syslogd e o arquivo `syslog`**: O `syslogd` é o demónio responsável pela gestão de logs no sistema, e o ficheiro `syslog` é um dos seus principais registos.

Exemplo Prático: Analisar os Logs Pode usar comandos como `tail` para ver as últimas entradas de um ficheiro de log ou `grep` para procurar por mensagens específicas.

```
$ tail -f /var/log/messages
$ grep "sshd" /var/log/auth.log
```

Exercício Prático: Rastrear um Evento 1. Force um erro (por exemplo, ao tentar iniciar um serviço com a sintaxe incorreta). 2. Use o comando `tail` ou `grep` para encontrar a mensagem de erro no ficheiro `/var/log/messages` ou `/var/log/syslog` e identifique o motivo do erro.