

Conteúdos

1. Serviços de rede

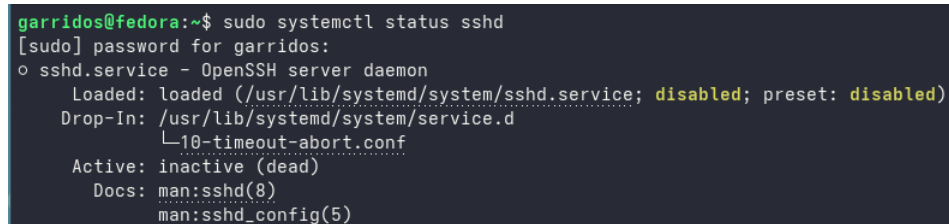
Nesta secção, vamos explorar como os serviços de rede são geridos no Linux, desde o seu início até ao encerramento, e os principais ficheiros e diretórios envolvidos neste processo.

- **Conceito Chave: O Papel do `systemd`**

O `systemd` é o gestor de sistema e de serviços padrão na maioria das distribuições Linux modernas, substituindo sistemas mais antigos como o SysVinit. Ele usa "unidades" para gerir processos, o que lhe dá um controlo mais granular, robusto e eficiente sobre os serviços. Em vez de scripts de inicialização simples, as unidades `systemd` podem definir dependências, o que garante que os serviços iniciam na ordem correta.

- **Exemplo Detalhado: Análise do Estado de um Serviço**

Vamos usar o comando `systemctl status` para obter informações detalhadas sobre o serviço de SSH (`sshd`). Uma imagem da saída deste comando pode ilustrar bem a informação disponível.



```
garridos@fedora:~$ sudo systemctl status sshd
[sudo] password for garridos:
○ sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; disabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
            └─10-timeout-abort.conf
   Active: inactive (dead)
     Docs: man:sshd(8)
           man:sshd_config(5)
```

Figura 1: Exemplo da saída do comando `systemctl status sshd`. Fonte: garridos.

A resposta mostra o estado atual do serviço (ativo/inativo), o PID (Process ID), há quanto tempo está a correr e as últimas linhas dos seus logs. Isto é crucial para diagnosticar rapidamente se um serviço está a funcionar.

- **Dicas de Resolução de Problemas**

- **Verificação de Logs**: Se um serviço não inicia, o primeiro passo é verificar os seus logs. O comando `journalctl -u [serviço]` mostra todo o histórico de logs do serviço, o que pode revelar a causa do erro.
- **Estado de Ativação**: Use `systemctl is-enabled [serviço]` para verificar se um serviço está configurado para iniciar automaticamente no arranque do sistema.

- **Exercício de Consolidação 1**. Inicie e verifique o estado do serviço web Apache (`httpd`). 2. Habilite o Apache para iniciar automaticamente no próximo arranque do sistema.

- **Para Aprofundar** Explore a diferença entre os comandos `systemctl start`, `restart` e `reload`. Investigar a estrutura de um ficheiro de unidade `service` em `/etc/systemd/system/` também é um excelente próximo passo.

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, como o `telnet` ou `ftp`. Ele espera por pedidos de conexão numa porta específica e, quando um pedido chega, inicia o serviço correspondente.

- **Conceito Chave: Servidor "On-Demand"**

O `xinetd` economiza recursos do sistema, pois os serviços geridos por ele só correm quando necessário, em vez de estarem sempre ativos.

- **Exemplo Detalhado: Configuração de um Serviço**

Vamos analisar o ficheiro de configuração do serviço `telnet` em `/etc/xinet.d/telnet`.

```

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

```

A linha `disable = yes` é a chave: para ativar o serviço, deve ser alterada para `no`.

- **Dicas de Resolução de Problemas**

- Se um serviço gerido por `xinetd` não funciona, verifique o ficheiro de configuração correspondente em `/etc/xinet.d/` para ter a certeza de que a opção `disable` está definida como `no`.

- **Exercício de Consolidação**

1. Encontre o ficheiro de configuração do serviço `ftp` no diretório `/etc/xinet.d/`.
2. Altere o valor da opção `disable` para habilitá-lo.
3. Reinicie o serviço `xinetd` para que as alterações entrem em vigor.

—

3. TCPWrappers

O `TCPWrappers` é uma ferramenta de segurança simples mas eficaz para controlar o acesso a serviços de rede, atuando como um "firewall de nível de serviço". Ele permite a criação de regras de acesso (permitir/negar) baseadas em endereços IP ou nomes de host.

- **Conceito Chave: O Ciclo `hosts.allow` -> `hosts.deny`**

As regras são processadas numa ordem específica: o sistema verifica primeiro o ficheiro `/etc/hosts.allow`. Se uma regra corresponder, o acesso é concedido e o `hosts.deny` é ignorado. Se não houver correspondência, o sistema verifica o `hosts.deny`. Se uma regra corresponder, o acesso é negado.

- **Exemplo Detalhado: Regras de Acesso**

Em `/etc/hosts.allow`:

```
sshd: 192.168.1.100
```

Em `/etc/hosts.deny`:

```
sshd: ALL
```

Neste exemplo, apenas a máquina com o IP `192.168.1.100` pode aceder ao serviço SSH. Todos os outros são bloqueados pela regra em `hosts.deny`.

- **Dicas de Resolução de Problemas**

- Cuidado com a ordem das regras! Uma regra ampla em `hosts.allow` pode anular regras mais específicas em `hosts.deny`.

- **Exercício de Consolidação**

1. Adicione uma regra para permitir o acesso ao serviço `ftp` a partir de um IP específico.
2. Adicione uma segunda regra para negar o acesso a `ftp` a todos os outros IPs, exceto ao endereço do passo 1.

—

4. NIS

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

- **Conceito Chave: Autenticação Centralizada**

O NIS permite que um utilizador inicie sessão em qualquer máquina cliente na rede com as mesmas credenciais, pois as informações da conta são geridas num servidor central (o servidor NIS).

- **Exemplo Detalhado: Listar Utilizadores NIS**

Após a configuração do cliente, podemos listar os utilizadores do servidor NIS com o comando `yycat`.

```
$ yycat passwd
```

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

- **Exercício de Consolidação**

1. Use o comando `ypwhich` para verificar a que servidor NIS a sua máquina cliente está conectada.
2. Use o comando `yycat` para listar as contas de grupo disponíveis.

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 Chave: O Processo DORA**

O DHCP funciona através de um processo de quatro etapas: **D**iscover (descoberta), **O**ffer (oferta), **R**equest (pedido) e **A**cknowledge (confirmação). Uma imagem pode ilustrar bem este fluxo.

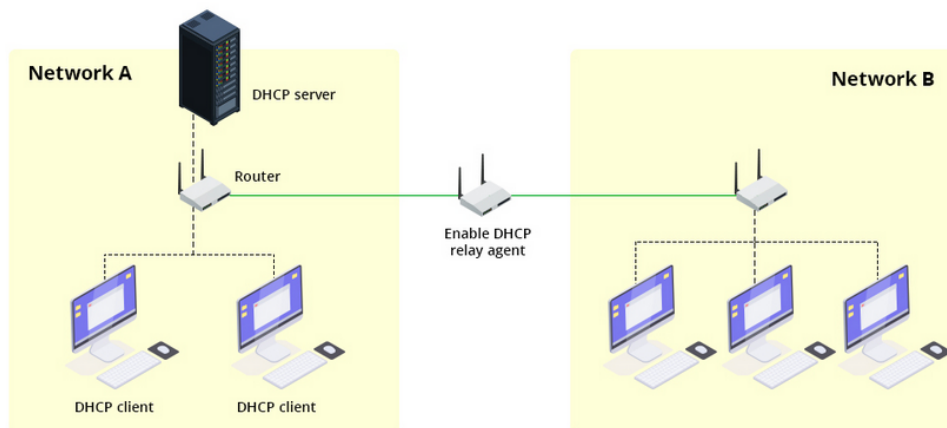


Figura 2: Diagrama do processo DORA (Discover, Offer, Request, Acknowledge). *Fonte:* <https://www.manageengine.com/products/oputils/images/network-a-b.jpg>.

- **Exemplo Detalhado: Configuração de uma Sub-rede**

Um exemplo de uma configuração no ficheiro `/etc/dhcp/dhcpd.conf` 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;  
}
```

- **Dicas de Resolução de Problemas**

- Para IPs estáticos, use a diretiva `host` para garantir que uma máquina específica receba sempre o mesmo IP.
- Se o cliente não recebe um IP, verifique os logs do servidor DHCP para ver se há erros de comunicação.

- **Exercício de Consolidação**

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.

- **Conceito Chave: Resolução de Nomes**

Quando um utilizador digita um nome de domínio, o cliente DNS consulta o servidor DNS para obter o endereço IP correspondente, permitindo que a conexão seja estabelecida. Este processo é hierárquico, começando pelos servidores de raiz e descendo até ao servidor autoritativo para o domínio em questão. Uma imagem pode ilustrar bem esta hierarquia.

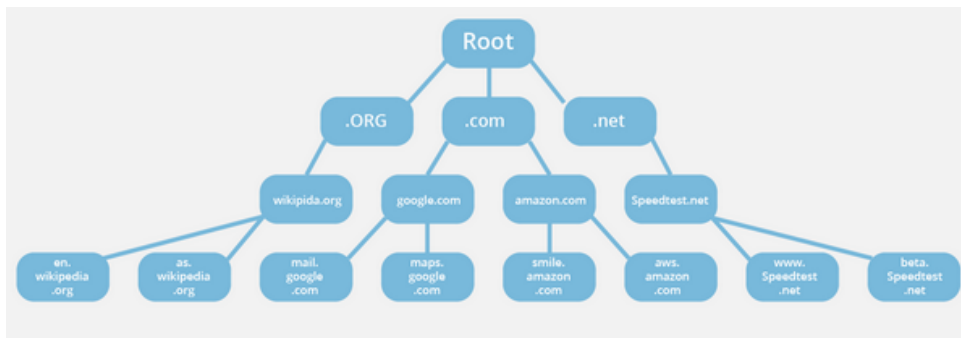


Figura 3: Fluxo de uma consulta DNS típica, mostrando a hierarquia de servidores. *Fonte: <https://www.cloudflare.com/img/learning/dns/glossary/dns-root-server/dns-root-server.png>.*

- **Exemplo Detalhado: Ficheiro de Zona**

Um ficheiro de zona simples para `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
```

- **Dicas de Resolução de Problemas**

- Use ‘dig’ ou ‘nslookup’ para testar a resolução de nomes. Se o ‘dig’ não funcionar, o problema pode estar na configuração do cliente ou do servidor.

- **Exercício de Consolidação**

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. 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.

- **Conceito Chave: Onde Encontrar Informação**

O diretório `/var/log` é o local central para a maioria dos logs do sistema e de aplicações. Cada ficheiro tem uma função específica (ex: `messages` para logs gerais, `auth.log` para autenticação).

- **Exemplo Detalhado: Analisar os Logs**

Use `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 de Consolidação**

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` e identifique o motivo do erro.