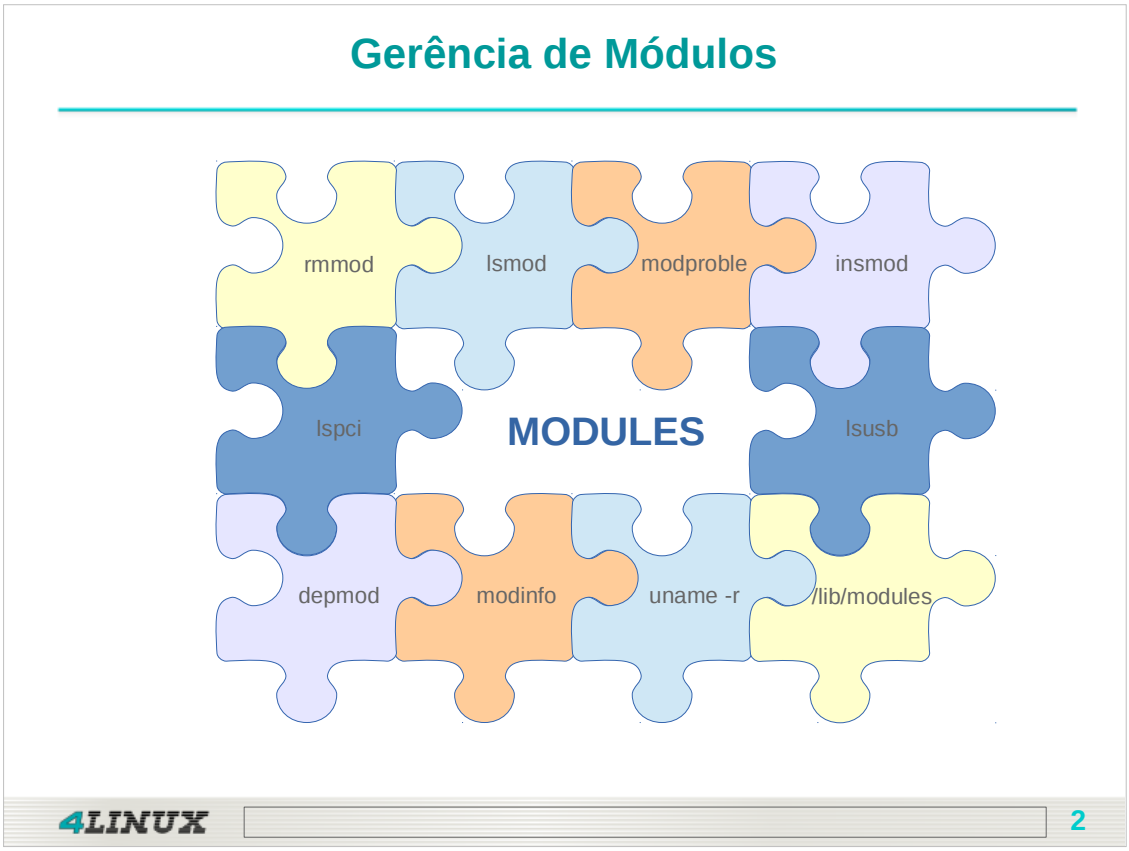
Curso 4451

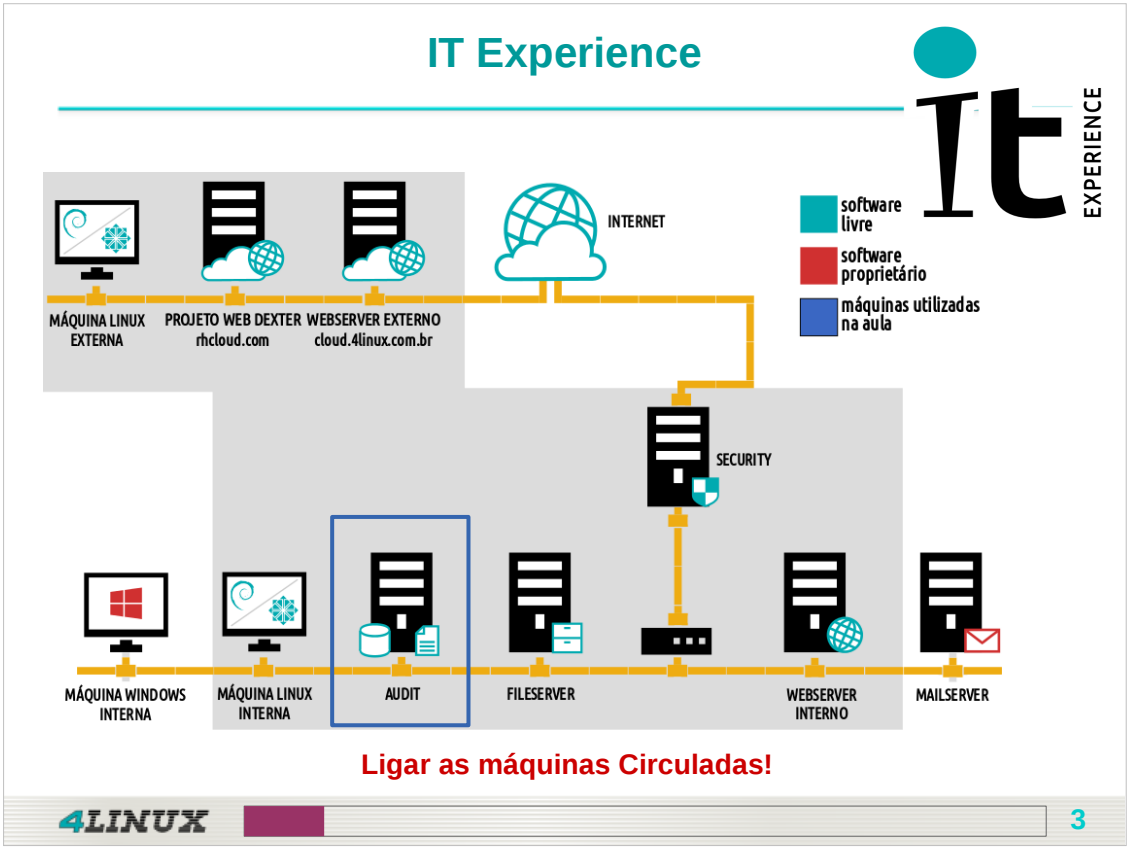
Linux Security Administration in Cloud



Fundamentação

Quando instalamos um Debian, RedHat, Suse, Slackware, entre outras distribuições, estamos utilizando um kernel que foi compilado pelos desenvolvedores da distribuição.

O kernel que vem por padrão em uma distribuição, deve ser capaz de rodar em praticamente qualquer PC e dar suporte a quaisquer tipos de recursos que o usuário pretenda utilizar, o desenvolvedor compila um kernel que fornece todas as funcionalidades básicas e, em separado, compila pedaços de código que dão suporte a funcionalidades mais específicas. Esses pedaços de código são os chamados módulos.


IT Experience
IT EXPERIENCE
INTERNET
software livre
software proprietário
máquinas utilizadas na aula
MÁQUINA LINUX EXTERNA
PROJETO WEB DEXTER rhcloud.com
WEBSERVER EXTERNO cloud.4linux.com.br
SECURITY
MÁQUINA WINDOWS INTERNA
MÁQUINA LINUX INTERNA
AUDIT
FILESERVER
WEBSERVER INTERNO
MAILSERVER
Ligar as máquinas Circuladas!
4LINUX
3

Anotações:

Objetivos da Aula

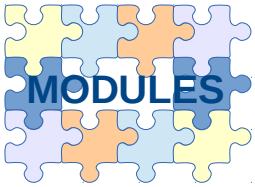
Aula 10

- Diferenciar módulos Built in de Módulos Externos;
- Listar módulos em um sistema GNU/Linux;
- Carregar e descarregar módulos do sistema.

Anotações:

Gerência de Módulos

- Um módulo é um trecho de código que oferece suporte a algum hardware ou funcionalidade;
- Podem ser do próprio kernel ou de terceiros;
- Além disso, quando compilamos ou recompilamos um kernel, os módulos podem ser “built-in” ou externos.



Anotações:

Gerência de Módulos

Servidor: Audit

1 - Vericando Módulos Buil-In:

```
1# uname -r
   3.2.0-4-486
2# cd /boot
3# grep -i quota config-3.2.0-4-486
   CONFIG_QUOTA=y
```

2 – Verificando Módulos Externos:

```
4# grep -i vfat config-3.2.0-4-486
   CONFIG_VFAT_FS=m
5# ls /lib/modules/
```

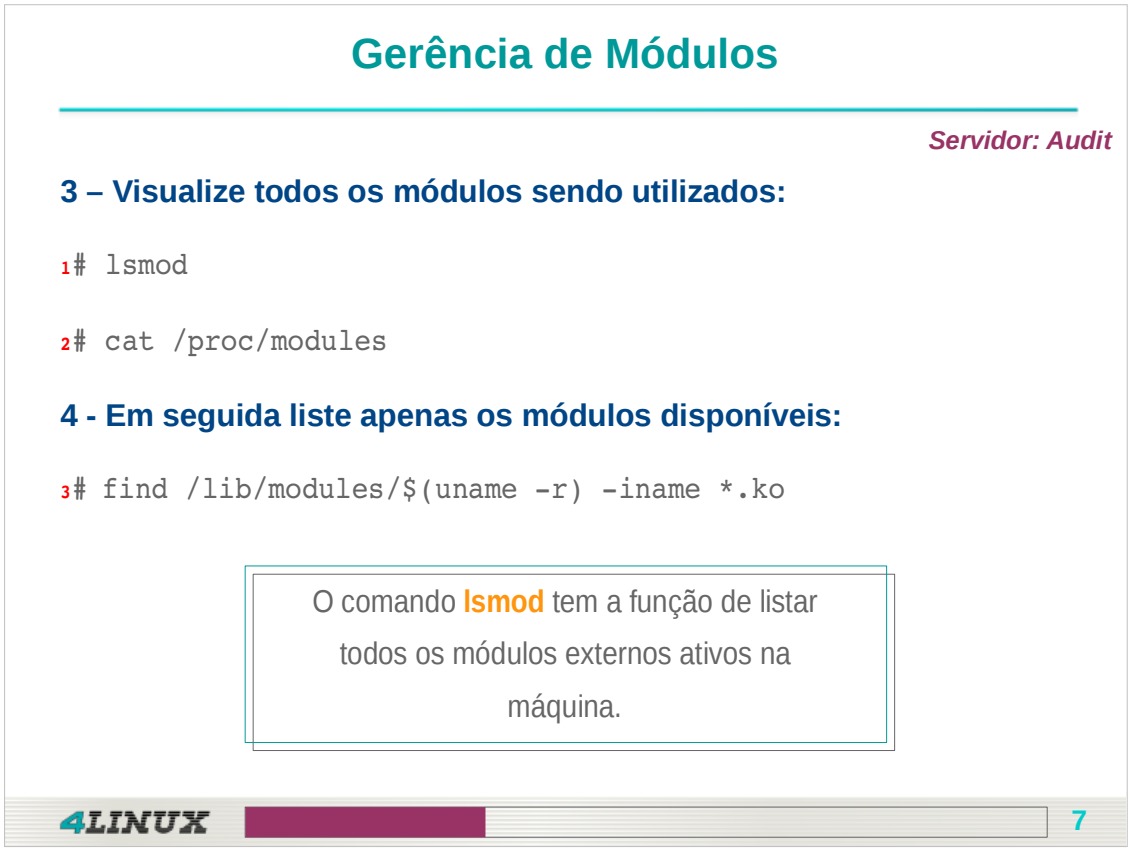
O arquivo **config-VERSÃO** guarda as informações do kernel que foi compilado no Sistema, informando os módulos buit-in e os módulos externos que ficam armazenados no diretório /lib/modules/VERSÃO.

Y = Buit-In

M = Módulo Externo

/lib/modules

Os módulos disponíveis, em geral, encontram-se no diretório “/lib/modules”, Diversos componentes do kernel do Linux são implementados como módulos, por exemplo, filesystems, device drivers, e novas camadas de protocolos de comunicação.

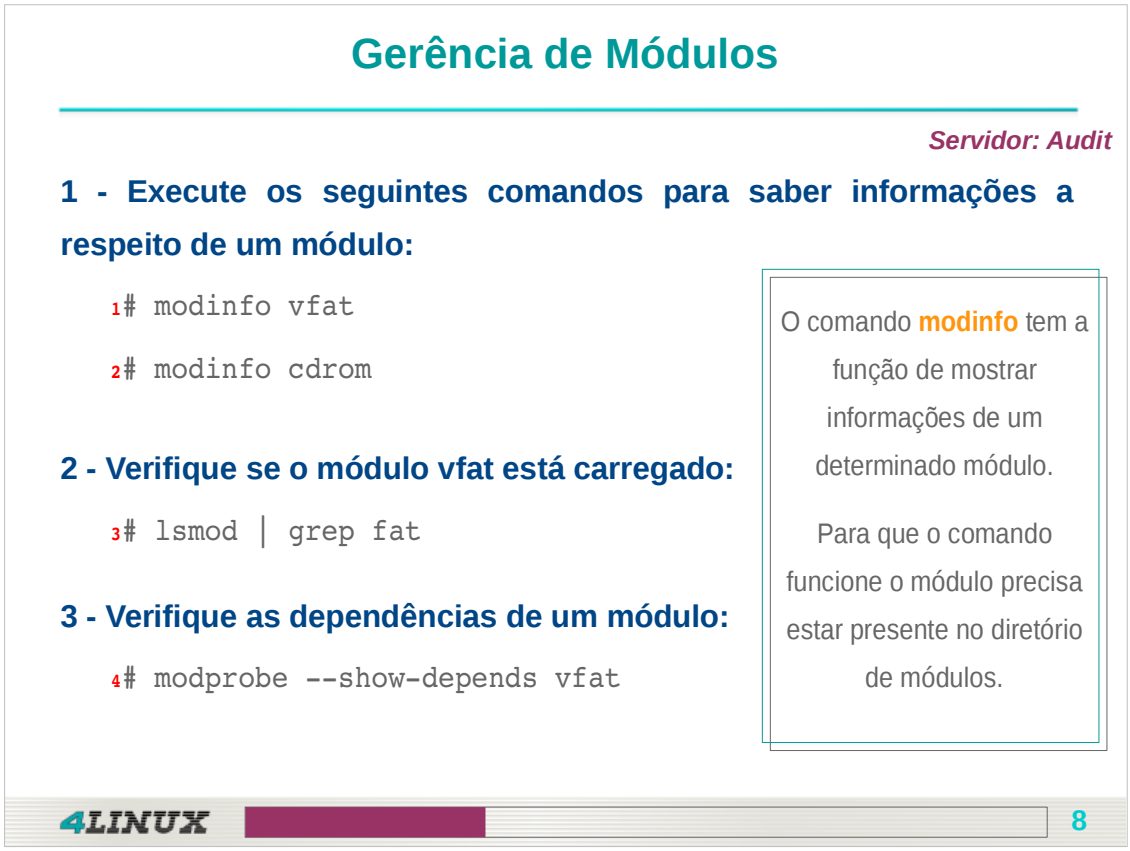


Listando Módulos

Ao executar o comando lsmod você pode ver quais módulos estão carregados atualmente no seu kernel. Veja que a saída do comando lsmod é em colunas, é listado todos módulos que estão carregados em memória, inclusive os que não estão em uso.

Onde:

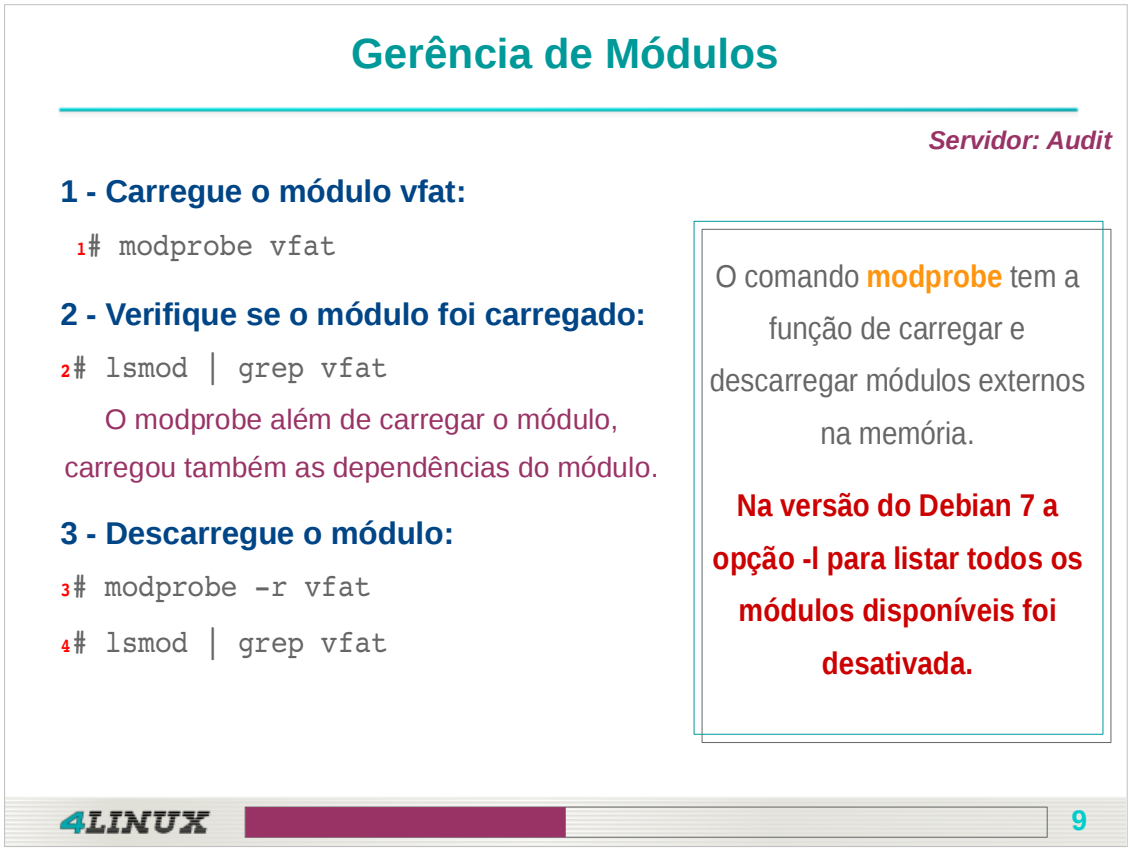
Module → Exibe o nome do módulo
Size → Exibe em bytes, o tamanho da memória do módulo
Used by → Exibe a contagem de quantas instâncias do módulo estão carregadas e o módulo que está usando; os valores são importantes porque não podemos remover um módulo que esteja sendo usado, a não ser que nesse campo, o valor seja zero.



Dependências

Algumas vezes, um módulo depende de outro para realizar determinadas operações. Se o módulo B depende de A, este deve ser carregado antes de ser possível carregar B.

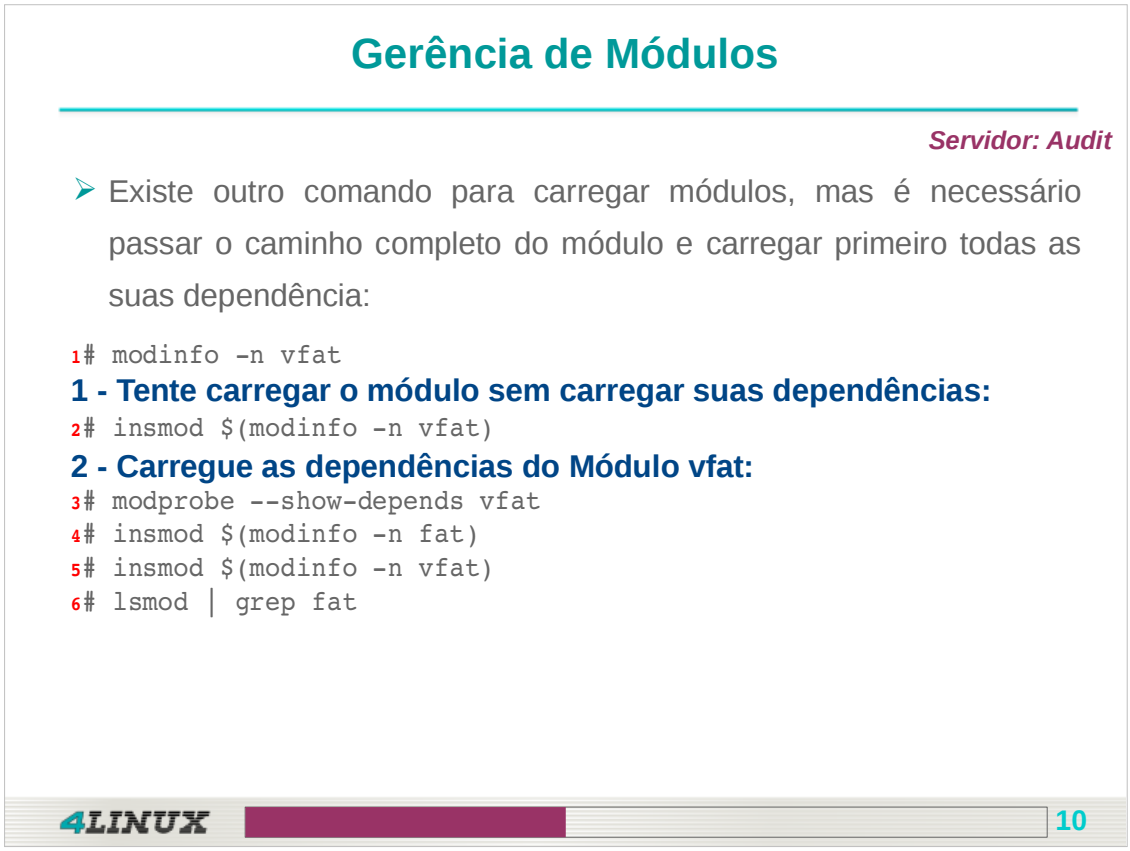
O usage counter de A é incrementado sempre que um módulo que depende dele é carregado. Deste modo, não se permite que A seja removido antes de seus dependentes.



Carregando Módulos

O kernel provê o comando modprobe para facilitar o gerenciamento de dependências.

Este comando tenta carregar automaticamente qualquer dependência do módulo solicitado. Por exemplo, ao tentar carregar o módulo MS-DOS, o comando modprobe carrega primeiro o módulo fat, seguido por MS-DOS.



Insmod

Os módulos também podem ser carregados através do programa insmod que crua uma estrutura do tipo module alocada quando seu carregamento é solicitado.

Esta estrutura contém símbolos globais que podem ser vistos pelo kernel e outros módulos, informando os pontos de entrada de suas funções, suas variáveis globais, seu usage counter, flags, entre outros.

Gerência de Módulos

Servidor: Audit

- Para descarregar o módulo individualmente temos o comando **rmmod**;
- Este comado somente removerá o módulo se ele não estiver sendo utilizado por outro módulo. É possível forçar com a opção **"-r",** mas tome cuidado ao utilizar esta funcionalidade e parar determinada aplicação.

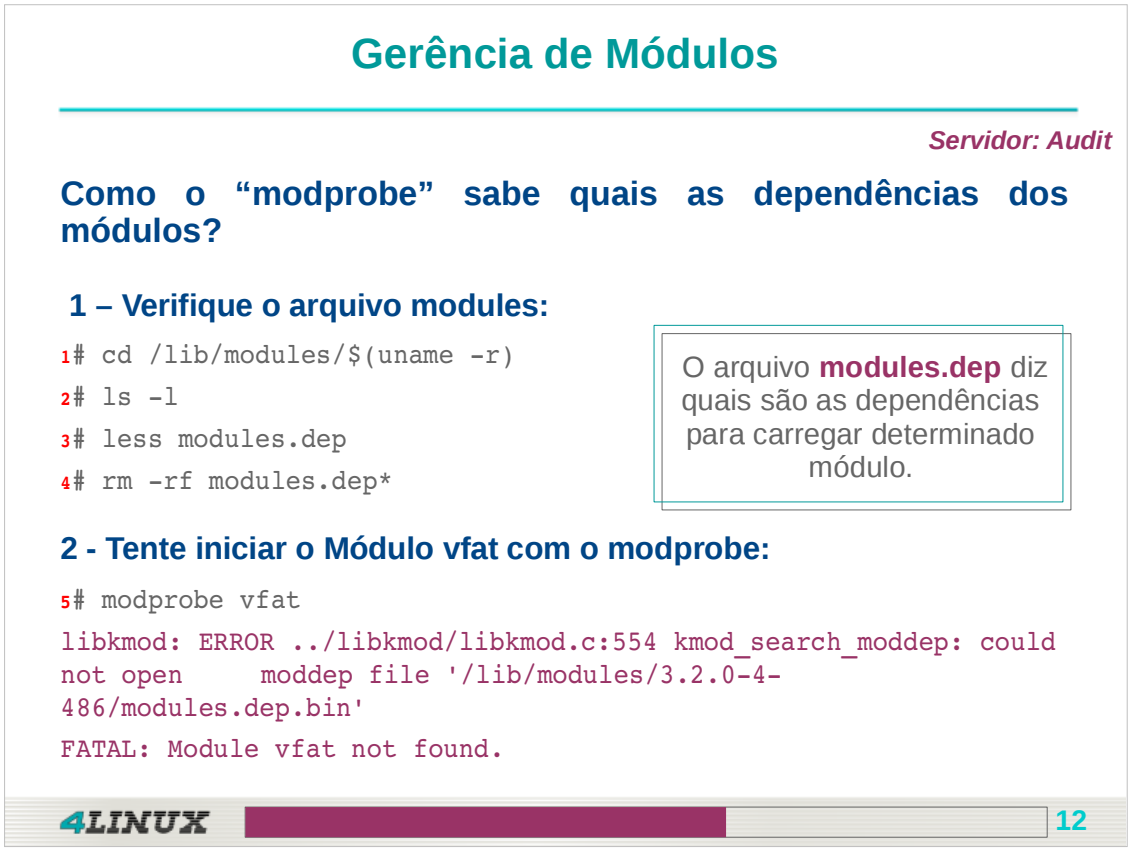
1 – Teste a remoção de módulos:

```
1# rmmod  vfat
```

Repare que ele não remove as dependências:

```
2# lsmod | grep fat
3# rmmod fat
4# lsmod | grep fat
```


Anotações:



O arquivo modules.dep

O comando modprobe faz uso de um arquivo chamado modules.dep para determinar as dependências de todos os módulos compilados para o kernel corrente.

Este arquivo é gerado pela execução, no start-up da máquina, de outro programa chamado depmod.

Ele avalia, durante o carregamento inicial do kernel, todos os módulos compilados, normalmente armazenados em /lib/modules, e gera o arquivo modules.dep.

Gerência de Módulos

Servidor: Audit

3 - Para gerar o arquivo novamente faça:

```
1# depmod   ou  depmod -v  (Verbose)
2# ls -l
```

4 - Tente iniciar o módulo vfat agora que o arquivo modules.dep foi gerado:

```
3# modprobe vfat
4# lsmod | grep fat
```


Anotações:

Gerência de Módulos

Servidor: Audit

1 - Iniciando módulos no boot da máquina:

```
1# vim /etc/modules
   vfat
```

2 - Criando alias de módulo:

```
2# vim /etc/modprobe.d/alias.conf
   alias windows vfat
```

<Reinicie o Servidor para Validar as alterações do slide anterior>

```
3# init 6
```


Anotações:

Gerência de Módulos

Servidor: Audit

1 - Validando módulos iniciado no boot:

```
1# lsmod | grep fat
```

2 - Validando o alias do módulo:

```
2# modprobe -r vfat
3# lsmod | grep  fat
4# modprobe windows
5# lsmod | grep vfat
```

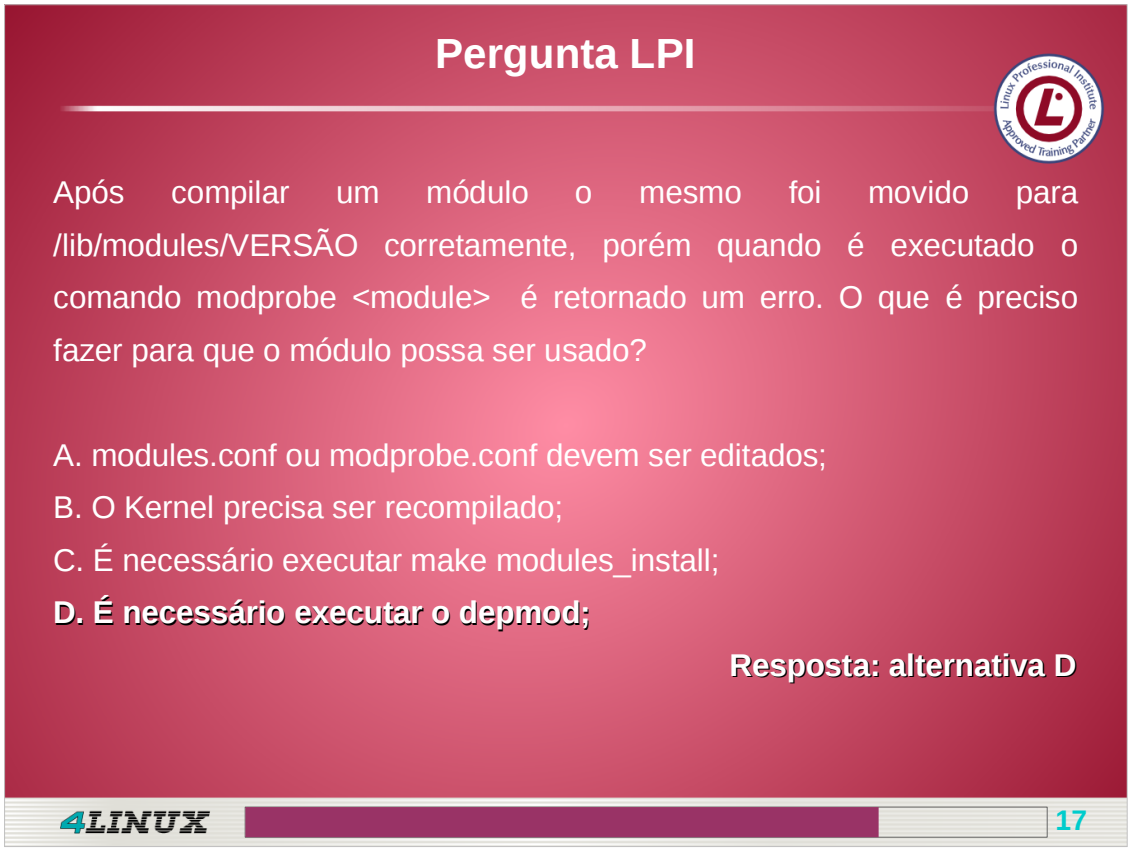

Anotações:

Pergunta LPI

Após compilar um módulo o mesmo foi movido para /lib/modules/VERSÃO corretamente, porém quando é executado o comando modprobe <module> é retornado um erro. O que é preciso fazer para que o módulo possa ser usado?

A. modules.conf ou modprobe.conf devem ser editados;
B. O Kernel precisa ser recompilado;
C. É necessário executar make modules_install;
D. É necessário executar o depmod;

Anotações:



Alternativa D: RESPOSTA CORRETA!

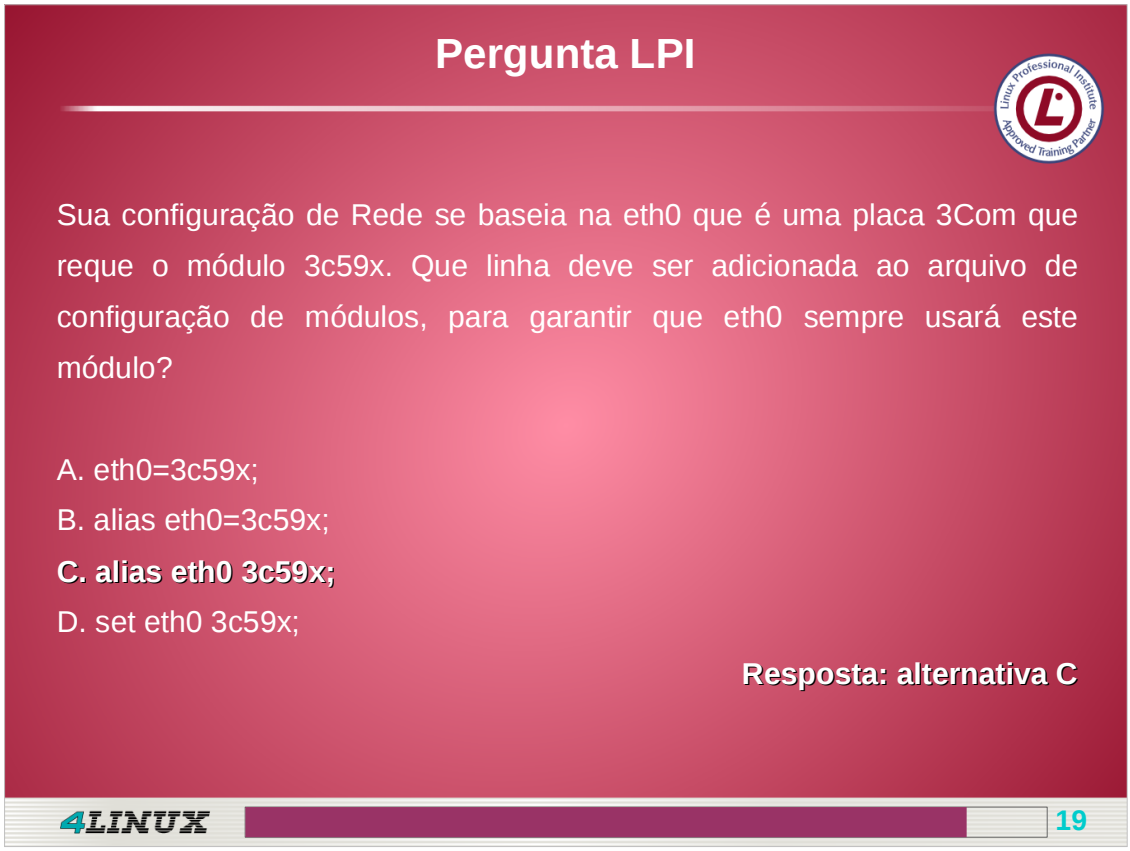
Deve-se executar o comando depmod toda vez que alterações forem feitas em relação aos módulos, Este comando produzirá um novo arquivo contendo as dependências dos módulos do kernel a serem instalados.

Pergunta LPI

Sua configuração de Rede se baseia na eth0 que é uma placa 3Com que reque o módulo 3c59x. Que linha deve ser adicionada ao arquivo de configuração de módulos, para garantir que eth0 sempre usará este módulo?

A. eth0=3c59x;
B. alias eth0=3c59x;
C. alias eth0 3c59x;
D. set eth0 3c59x;

Anotações:



Alternativa C: RESPOSTA CORRETA!

Para que um módulo possa ser invocado a partir de um apelido é necessário que ele seja adicionado ao arquivo /etc/modprobe.d/alias.conf conforme a sintaxe descrita na alternativa C.

Próximos Passos

Para que você tenha um melhor aproveitamento do curso, participes das seguintes atividades disponíveis no Netclass:

- Executar as tarefas do **Practice Lab**;
- Resolver o **Desafio Appliance Lab** e postar o resultado no Fórum Temático;
- Responder as questões do **Teste de Conhecimento** sobre o conteúdo visto em aula.

Mãos à obra!