

Redes de Computadores II

Aula 08 - Autenticação e Compartilhamento de Arquivos – Parte III

Apresentação

Nas aulas passadas, você aprendeu sobre a necessidade de um serviço de autenticação e compartilhamento de arquivos e estudou sobre esses serviços.

Além disso, viu que o LDAP é um dos protocolos mais utilizados para implementar o serviço de autenticação. Sobre o LDAP, você aprendeu o protocolo e viu como instalar e configurar um servidor, mas não se preocupou muito com a finalidade para a qual ele seria utilizado.

Nesta aula, você vai aprender a usar o LDAP para criar o serviço de autenticação. Vai aprender também a configurar o serviço de compartilhamento de arquivos usando o protocolo NFS. Usaremos LDAP+NFS para que clientes Linux autenticuem e acessem arquivos de um servidor Linux.



Video 1 - Apresentação

Objetivos

Após o final desta aula, você será capaz de:

- Criar um serviço de autenticação usando LDAP, em que máquinas clientes Linux se autenticam em um servidor Linux.
- Criar um serviço de compartilhamento de arquivos no Linux usando NFS, de modo que os arquivos dos usuários que estão no servidor possam ser acessados da máquina do cliente.

Usando o LDAP para Autenticação

Vamos agora ver como utilizar um servidor LDAP para armazenar as informações sobre os usuários e autenticá-los quando tentarem utilizar uma máquina cliente.

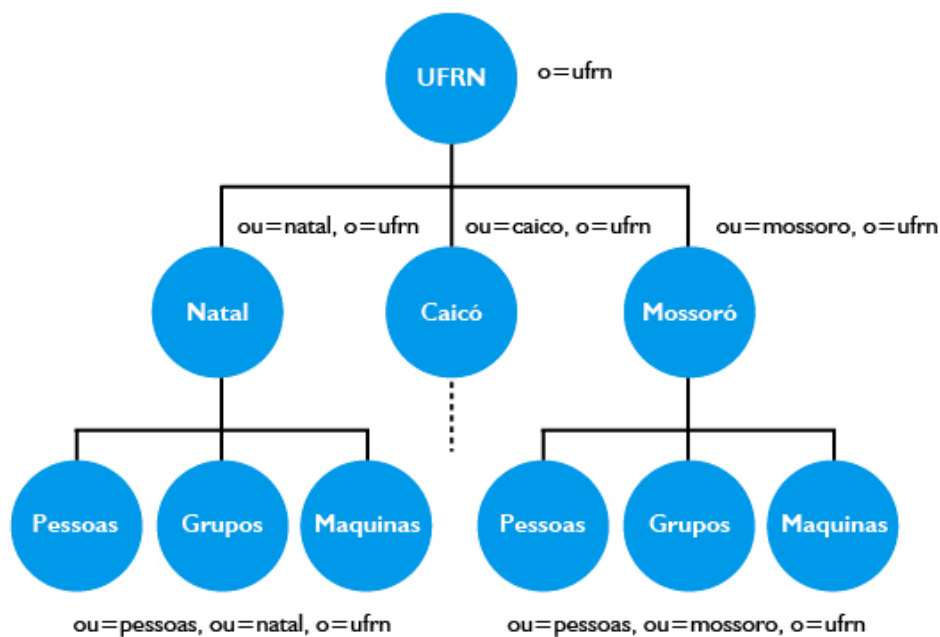
Estrutura da árvore LDAP para autenticação de usuários

Na aula passada, vimos um exemplo de uma árvore LDAP, na qual tínhamos um nó raiz do tipo “o”, para representar a organização, em que usamos “o=ufrn”. Todos os outros nós da árvore eram nós do tipo “ou” e representavam lugares ou cursos. Como você estudou, uma OU pode representar qualquer coisa.

Quando vamos usar o LDAP para autenticar usuários é muito comum criarmos mais três nós do tipo “ou” abaixo de cada nó que representa o lugar onde os usuários serão cadastrados. **Esses três novos nós são: “Pessoas”, “Grupos” e “Máquinas” e serão utilizados para armazenarem, respectivamente, os objetos que representam: os usuários, os grupos de usuários, e os computadores.**

Para esta aula, vamos usar como árvore LDAP, a árvore mostrada na Figura 1. Essa árvore é muito semelhante à utilizada na aula passada, com a diferença que eliminamos os nós referentes aos cursos. Portanto, os alunos de todos os cursos de cada unidade serão cadastrados no nó da unidade. Veja que só fizemos isso para a árvore ficar menor e simplificar a nossa explicação. Em um caso real, provavelmente deixaríamos os nós com os nomes dos cursos! Observe também que acrescentamos os três novos nós citados no parágrafo anterior abaixo do nó de cada unidade.

Figura 01 - Árvore LDAP usada nesta aula.



Para criar esses novos nós poderíamos fazer o seguinte.

Vamos assumir para o texto desta aula que um servidor LDAP está instalado (com a árvore mostrada na Figura1) em uma máquina com endereço IP 10.1.1.1. Desse modo, sempre que você vir esse endereço em algum lugar nessa aula é porque o endereço IP do servidor LDAP deve ser informado.

Simplificando a Gerência de Usuários e Grupos

Na aula passada, você também aprendeu a inserir usuários na sua base LDAP. Viu como fazer isso criando um arquivo *ldif* com as informações dos usuários e usando o programa *ldapadd* para ler esse arquivo e inserir as informações. Existe uma forma muito mais simples de manipular usuários e grupos, que é utilizando os programas de um pacote chamado *ldapscripts*. A seguir, descreveremos como realizar essa tarefa.

1. Para instalar esse pacote, digite:

```
apt-get install ldapscripts
```



Video 02 - Autenticação LDAP



Video 03 - LDAP Parte 1

Os programas do pacote `ldapscripts` utilizam um arquivo de configuração chamado `ldapscripts.conf`, que fica no diretório `/etc/ldapscripts`.

2. Edite o arquivo `/etc/ldapscripts/ldapscripts.conf` de acordo com a configuração do seu servidor LDAP. Na Figura 2, mostramos os valores usados para nosso servidor LDAP de exemplo.

```
1 SERVER=10.1.1.1
2 BINDDN='cn=admin,o=ufrn'
3 BINDPWDFILE="/etc/ldapscripts/ldapscripts.passwd"
4 SUFFIX='ou=natal,o=ufrn'
5 GSUFFIX='ou=Grupos'
6 USUFFIX='ou=Pessoas'
7 MSUFFIX='ou=Maquinas'
8 GIDSTART=10000
9 UIDSTART=10000
10 MIDSTART=10000
11 PASSWORDGEN="<ask>"
```

Figura 2 - Arquivo `ldapscripts.conf`

Os três primeiros parâmetros informam, respectivamente, o IP do servidor LDAP, o usuário administrador, e um arquivo que contenha a senha desse usuário. Explicaremos em breve como criar esse arquivo.

Os parâmetros `GSUFFIX`, `USUFFIX`, e `MSUFFIX`, indicam, respectivamente, os nomes dos nós em que os grupos, os usuários, e as máquinas serão inseridos. O valor do parâmetro `SUFFIX` será concatenado aos parâmetros anteriores para

formar o nome completo daqueles nós. Ou seja, um usuário, por exemplo, será inserido no nó “ou=pessoas,ou=natal,o=ufrn”, e um grupo será inserido em “ou=grupos,ou=natal,o=ufrn”.

Os parâmetros GIDSTAR, UIDSTART, e MIDSTAR, indicam, respectivamente, os identificadores numéricos iniciais que serão atribuídos aos grupos, usuários, e as máquinas. O primeiro usuário, por exemplo, terá o identificador 10000, o segundo usuário 10001, e assim sucessivamente.

A última linha fará com que durante o cadastro de novos usuários seja solicitada uma senha para ele.

Para criar o arquivo contendo a senha do administrador do LDAP (cn=admin,o=ufrn), digite os comandos a seguir, substituindo **senha** pela senha desse usuário. Essa senha é a mesma senha que foi colocada no parâmetro olcRootDN do arquivo de configuração do servidor (veja Figura 13 da Aula 9).

```
sudo sh -c "echo -n 'senha' > /etc/ldapscripts/ldapscripts.passwd"
sudo chmod 400 /etc/ldapscripts/ldapscripts.passwd
```

3. Agora basta utilizar um dos programas do pacote `ldapscripts` para manipular usuários e grupos. Os comandos devem ser digitados pelo usuário *root*, pois eles precisam ler o arquivo `ldapscripts.passwd`. O Quadro1 mostra exemplos dos comandos mais utilizados e o seu significado.

Comando	Significado
<code>ldapadduser maria alunos</code>	Cria o usuário <i>maria</i> e o insere no grupo <i>alunos</i>
<code>ldapsetpasswd maria</code>	Troca a senha do usuário <i>maria</i>
<code>ldapdeleteuser maria</code>	Exclui o usuário <i>maria</i>
<code>ldapaddgroup teste</code>	Cria o grupo teste

Comando	Significado
<code>ldapdeletegroup teste</code>	Exclui o grupo teste
<code>ldapaddusertogrouppl maria teste</code>	Insere o usuário <i>maria</i> no grupo teste

Quadro 1 - Comandos do pacote `ldapscripts`

Saiba que o comando para criação de usuários na LDAP não cria a pasta do usuário. Isso precisa ser feito manualmente pelo usuário *root*.

Atividade 01

1. Para que serve o pacote `ldapscripts`?
2. Uma forma de saber mais sobre cada comando é usando o `man` (manual do programa). Use o `man` para saber mais detalhes de cada um dos comandos acima. Ex.: No Linux, em um terminal, execute: `man ldapadduser`.

Criando as Pastas dos Usuários no Servidor

Lembre-se que ao criar os usuários no LDAP, as pastas deles não são criadas. Portanto, é necessário que elas sejam criadas manualmente. Para isso, basta utilizar o comando `mkdir`. Por exemplo, segue o comando para criar a pasta de *Maria*.

```
sudo mkdir /home/maria
```

Depois disso, é necessário trocar o proprietário da pasta e o grupo para os referentes a Maria. Supondo que *maria* pertence ao grupo *alunos*, o comando seria.

```
sudo chown -R maria:alunos /home/maria
```

Mas, atenção! A máquina onde estão os arquivos de Maria precisa estar configurada para autenticar no servidor LDAP. A confusão é que quando esta máquina é o próprio LDAP, costuma-se achar que ela já está configurada para isso. Mas não é verdade! Ela precisa ser configurada para autenticar no LDAP (que é ela mesma). A configuração é a mesma que mostraremos mais adiante, nesta aula para as máquinas clientes. Afinal, ela é um cliente dela mesma.

Parece confuso, mas você precisa pensar em termos de programas e não de máquinas. Os *programas* da máquina que controlam os acessos aos arquivos, precisam das informações do LDAP, e eles não sabem que o *programa servidor* LDAP está na própria máquina.

LDAP como Servidor de Autenticação de Usuários

Finalmente, agora vamos ver como criamos um servidor de autenticação usando o LDAP. Na verdade, o lado do servidor está pronto! Ou seja, tudo que tínhamos de fazer na máquina em que o programa slapd executa já foi feito, pois para o LDAP não interessa para que ele será usado. Todo LDAP tem uma árvore definida e objetos criados nessa árvore. A única coisa que fizemos especificamente para o serviço de autenticação foi criar os nós “pessoas”, “grupos”, e “máquinas”, e inserir os objetos nesses nós.

Portanto, a configuração do serviço de autenticação vai se concentrar nas máquinas clientes, ou seja, nas máquinas que vão autenticar os usuários remotamente usando o LDAP. A primeira coisa a fazer em cada máquina cliente é instalar os pacotes *libnss-ldap* e *libpam-ldap*, conforme mostrado a seguir.

```
apt-get install libnss-ldap libpam-ldap
```

Durante a instalação desses pacotes, será executado um programa de configuração que faz sete perguntas. Essas perguntas, e as respostas que deveriam ser dadas para nosso exemplo (texto em **negrito**), são mostradas a seguir.

- 1ª - Qual o protocolo e endereço IP do servidor LDAP? **ldap://10.1.1.1/**
- 2ª - Nó da árvore a partir do qual os usuários serão pesquisados? **ou=natal,o=ufrn**
- 3ª - Versão do protocolo LDAP a ser utilizada? **3**
- 4ª - Tornar o usuário root local administrador do LDAP? **Sim**
- 5ª - O Diretório LDAP requer login? **Não**
- 6ª - Usuário administrador do LDAP? **cn=admin,o=ufrn**
- 7ª - Senha do usuário administrador do LDAP? **coloque-a-senha-do-admin-aqui**

Na verdade, quando se responde “Não” para a 5ª pergunta, a 6ª e a 7ª pergunta não deveriam ser feitas, mas, dependendo da distribuição Linux sendo utilizada, elas ainda são feitas.

Após responder as perguntas será criado o arquivo `ldap.conf` na pasta `/etc`, com o conteúdo mostrado na Figura 3. Há diversas outras linhas no arquivo que é gerado, mas elas estão todas comentadas (iniciam com `#`), portanto não são usadas!

```
1 base ou=natal,o=ufrn
2 uri ldap://10.1.1.1/
3 ldap_version 3
4 pam_password clear
```

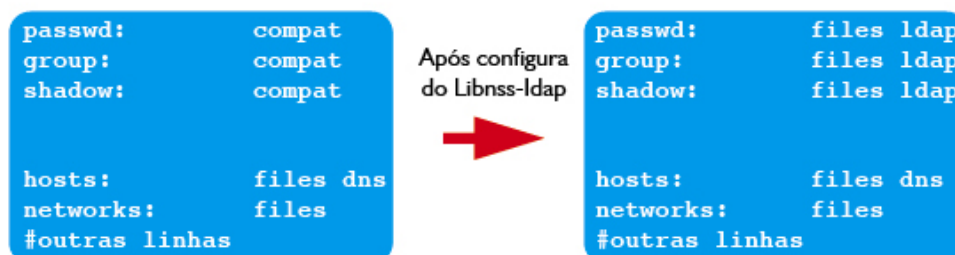
Figura 3 - Arquivo `/etc/ldap.conf`

Caso queira refazer essa configuração, ou seja, alterar algum valor que você informou, é possível editar o arquivo manualmente. Além disso, dependendo da distribuição Linux sendo utilizada, esse arquivo pode ficar na pasta `/etc/ldap` ou invés da pasta `/etc`.

Precisamos agora criar os arquivos `pam_ldap.conf` e `libnss-ldap.conf` que ficam, ambos, na pasta `/etc`. Embora esses arquivos possam conter diversas informações, tudo que eles precisam para que a autenticação funcione é do mesmo conteúdo mostrado para o arquivo `ldap.conf`. Portanto, basta criar esses dois arquivos com o conteúdo mostrado na Figura 3.

Agora precisamos dizer ao sistema operacional que use o ldap para a autenticação. Para isso, vamos alterar o conteúdo do arquivo `/etc/nsswitch.conf` para solicitar que ele também use o ldap para obter as informações dos nomes dos usuários, dos grupos, e as senhas dos usuários. Isso é feito alterando o valor das três primeiras linhas do arquivo. A Figura 4 mostra o arquivo antes da alteração e depois.

Figura 04 - Arquivo `/etc/nsswitch.conf` antes e depois da configuração do pacote `libnss-ldap`.



Para estar se sua máquina cliente está conseguindo obter as informações do LDAP, digite o comando `getent`, conforme mostrado a seguir, para obter informações de um usuário do LDAP. O comando mostrado obtém informações do usuário *maria* (assumindo que ele existe no ldap).

```
sudo getent passwd maria
maria:x:10001:10001:MariaSilva:/home/maria:/bin/bash
```

Outra forma de testar a autenticação é realizando um `ssh` na máquina cliente, para a própria máquina cliente, com um usuário que existe no ldap. O exemplo a seguir utiliza o usuário *maria*.

```
ssh maria@localhost
```

Se o login for bem sucedido é porque o cliente conseguiu autenticar no LDAP.

Atividade 02

1. Qual o arquivo de configuração em que se informa em qual servidor LDAP uma máquina cliente deve se autenticar?
 2. Veja o que acontece ao se digitar o comando `getent passwd` (sem especificar nenhum usuário).
-

Veja aqui a explicação em vídeo sobre como fazer uma máquina Linux autenticar em um servidorLDAP.



Video 04 - LDAP Parte 2

Confira aqui um tutorial de configuração de uma autenticação no Linux usando LDAP.



Video 05 - Tutorial de Autenticação no Linux

NFS

Agora vamos estudar o serviço de compartilhamento de arquivo, que conforme falamos antes, normalmente é utilizado em conjunto com o serviço de autenticação. Ou seja, não adianta muito o serviço de autenticação permitir que você utilize qualquer máquina da rede se seus arquivos não aparecerem lá.

Na sessão anterior, quando sugerirmos que *Maria* executasse um `ssh` para a própria máquina cliente, após o login iria aparecer a mensagem "*Could not chdir to home directory /home/maria: No such file or directory*". Isso significa que a pasta de *maria* (`/home/Maria`) não existe na máquina local. Ela conseguirá efetuar o login, mas o Shell iniciará com a pasta `/` (raiz)!

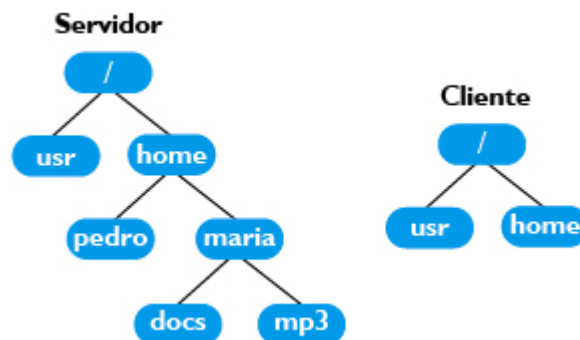
Nesta sessão, vamos aprender a usar o NFS (*Network File System*) para compartilhar pastas pela rede.

O que o NFS faz é associar uma pasta do cliente com uma pasta do servidor.

Desse modo, sempre que o cliente acessar essa pasta, na verdade ele estará acessando a pasta do servidor. Com isso, a pasta de Maria que está no servidor iria "aparecer" na máquina cliente.

A Figura 5 mostra como estariam os discos (e as pastas) na máquina servidora, na qual existem os arquivos de *Maria*, e na máquina cliente, em que não existe nada relacionado àquele usuário.

Figura 05 - Discos do servidor e do cliente antes do compartilhamento.



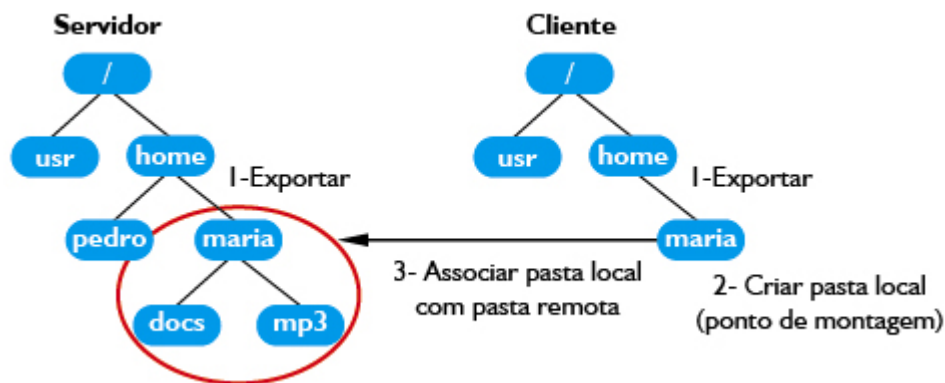
Para compartilhar as pastas de *Maria* são necessários três passos:

- O servidor precisa exportar a pasta de *maria* (`/home/maria`). Por exportar, entenda liberar o acesso a essa pasta (e suas subpastas) para as máquinas da rede.
- A máquina cliente precisa criar uma pasta local para onde a pasta remota será associada. Essa pasta local é chamada *ponto de montagem*.

- A máquina cliente precisa associar a pasta local com a pasta do servidor. Esse processo é chamado de “montar a pasta”.

A Figura 6 mostra como ficaria a configuração das máquinas após os três passos anteriores serem realizados. Nesse ponto, a máquina cliente acessa as pastas docs e mp3 de *maria* como se elas fossem pastas locais (da máquina cliente).

Figura 06 - Configuração após compartilhamento.



A pasta no cliente, em que é feita a associação, não precisa ser igual à pasta do servidor. No exemplo acima, a pasta /home/maria do servidor poderia, por exemplo, ter sido associada a uma pasta /usuários na máquina cliente.

Atividade 03

1. O que significa o termo *ponto de montagem* para o NFS?
2. O que significa o termo *montar uma pasta* para o NFS?
3. É possível apagar uma pasta NFS que está montada em uma máquina cliente?

Instalando e configurando o NFS

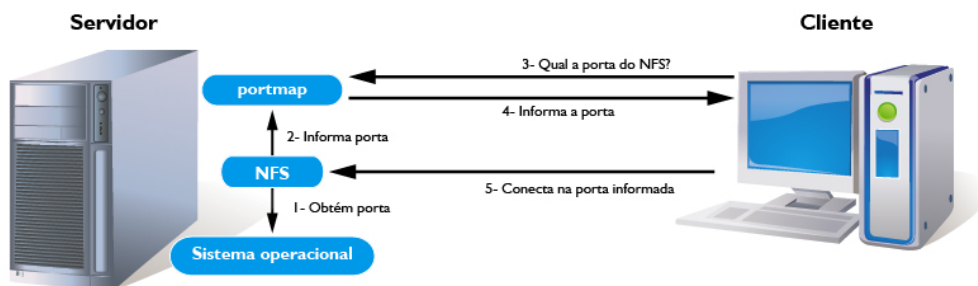
Para instalar o NFS é necessário instalar dois pacotes na máquina que vai atuar como servidora, ou seja, onde estão os arquivos e pastas a serem compartilhados. Para isso, digite o comando a seguir.

```
apt-get install nfs-kernel-server rpcbind
```

O Portmap

É importante saber que diferentemente da maioria dos programas que atuam como servidor, o NFS não escuta em uma porta fixa. Ou seja, enquanto um servidor web sempre escuta na porta 80, um servidor de e-mail na porta 25, e um servidor de DNS na porta 53, a cada vez que você executa NFS ele pode usar uma porta diferente. Portanto, a princípio, o programa cliente não tem como saber a porta onde o servidor está. Para resolver isso, o NFS informa a um programa chamado *portmap* em qual porta ele está escutando. Como o *portmap* sempre escuta em uma mesma porta (porta 111), os clientes sabem que porta é essa, e perguntam ao *portmap* em qual porta o NFS está escutando. A Figura 7 mostra os passos até que o cliente localize o servidor NFS.

Figura 07 - Localizando o servidor NFS através do *Portmap*



Do que foi dito no parágrafo anterior, o mais importante é que você se lembre que além do NFS, o *portmap* também tem que estar executando, senão o NFS não vai funcionar. Você não precisa iniciar o portmap manualmente, pois ele já é iniciado automaticamente. De qualquer modo, o comando para iniciá-lo é:

```
/etc/init.d/rpcbind start
```

Configurando o NFS

Vamos agora ver como realizar os três passos que citamos anteriormente para compartilhar uma pasta.

Passo 1.

Para exportar uma pasta basta inserir uma linha para ela no arquivo `/etc/exports`, conforme mostrado a seguir.

```
1 /home/maria *(rw,sync,root_squash,no_subtree_check)
```

Figura 8 - Arquivo `/etc/exports`

O `*` significa que qualquer máquina da rede pode montar essa pasta. Ao invés do asterisco poderia ter sido colocado um endereço IP, por exemplo. Nesse caso, apenas a máquina informada poderia montar a pasta.

O `"rw"` significa que o cliente pode ler e escrever na pasta. Se você desejasse que os clientes pudessem apenas ler o conteúdo da pasta, bastava informar `"ro"` ao invés de `"rw"`.

A opção `"sync"` significa que as operações solicitadas pelo cliente sejam executadas imediatamente.

A opção `"root_squash"` significa que o usuário `root` da máquina cliente não terá direito de root nessa pasta. Ou seja, ele se comportará como um usuário comum.

A opção `no_subtree_check` desativa algumas checagens que o servidor faz para ter certeza que o cliente está acessando um arquivo/pasta dentro da pasta exportada. Apesar de essa checagem aparentemente melhorar a segurança, pode-se gerar mais problemas que benefícios quando ocorrem muitas alterações na pasta exportada (criar/apagar arquivos e pastas). Por isso, é mais comum utilizar a opção `no_subtree_check` para desativar essas checagens.

Depois de inserir a linha mostrada no arquivo, basta executar o comando a seguir para ativar o compartilhamento.

```
exportfs -a
```

Se quiser ver as pastas compartilhadas atualmente, digite o mesmo comando sem o “-a”.

Além disso, para cancelar o compartilhamento, basta digitar o comando “exportfs -u */home/maria” (depois apague a linha do arquivo /etc/exports referente a pasta que não deseja mais compartilhar). O * informado nesse comando tem a mesma função que tinha na linha da Figura 8, ou seja, identificar os ips dos clientes.

Passo 2.

Crie o ponto de montagem na máquina cliente. Ou seja, crie a pasta local para a pasta remota a qual estará associada. Depois modifique o proprietário e o grupo dela. Esse dois procedimentos são mostrados a seguir.

```
mkdir /home/maria  
chown maria:alunos /home/maria
```

Naturalmente, essa máquina deve estar configurada para autenticar no servidor LDAP para poder reconhecer os usuários (no caso *maria*).

Passo 3.

A máquina cliente precisa instalar o programa a seguir.

```
apt-get install nfs-common
```

Depois, basta fazer a associação da pasta local com a pasta remota. Para isso, insira a linha a seguir no arquivo /etc/fstab da máquina cliente.

1	10.1.1.1:/home/Maria /home/Maria nfs rsize=8192,wsiz=8192,timeo=14,intr
---	---

Figura 9 - Linha a ser adicionada ao arquivo /etc/fstab.

O texto "10.1.1.1:/home/Maria" indica a pasta remota e a máquina onde ela está. O texto seguinte "home/Maria" é o nome da pasta local que ficará associada à pasta remota. Depois temos "nfs", que indica o tipo de sistema de arquivos. Os demais parâmetros são para melhorar o desempenho, além de não travar a máquina cliente caso a conexão com o servidor pare de funcionar.

Você pode usar sempre esses mesmos parâmetros, alterando, evidentemente, o IP do servidor e os nomes das pastas.

Quando a máquina for reiniciada, o arquivo /etc/fstab será lido e a pasta será montada. Entretanto, você provavelmente não vai querer reiniciar seu servidor só porque está compartilhando uma pasta.

Por isso, você pode executar o comando a seguir para ativar a configuração imediatamente, ou seja, para montar a pasta remota localmente, conforme indicado no arquivo /etc/fstab.

```
mount -a
```

Pronto! Agora pode acessar a pasta local /home/Maria que na verdade estará acessando a pasta do servidor.

Voltando ao exemplo que tínhamos utilizado para testar a autenticação do LDAP no cliente, Maria pode agora fazer ssh para a própria máquina cliente. Após o login, ela estará na sua pasta, e se ela digitar o comando ls, o resultado apresentado seria o mostrado a seguir.

Assim sendo, as pastas docs e mp3 que estão no servidor aparecem como se estivessem na máquina local.

```
ls /home/maria  
docs mp3
```

Atividade 04

1. Se o portmap não estiver executando, o NFS vai funcionar?
 2. Qual a função do comando “exportfs -a”?
 3. Acesse o endereço <http://www.vivaolinux.com.br/artigo/FSTAB-Sua-funcao-e-parametros> e veja qual a função do arquivo FSTAB do Linux.
-

Todos os Usuários

Em uma rede real, você tipicamente exportaria o /home do servidor e o montaria no /home das máquinas clientes. Isso faria com que os usuários pudessem utilizar qualquer máquina da rede.

Monitorando o NFS

Vamos ver agora mais alguns comandos importantes quando utilizamos o NFS. Nos exemplos, assumo que a máquina que está atuando como servidor NFS se chama Máquina-A.

Programas que compõem o NFS

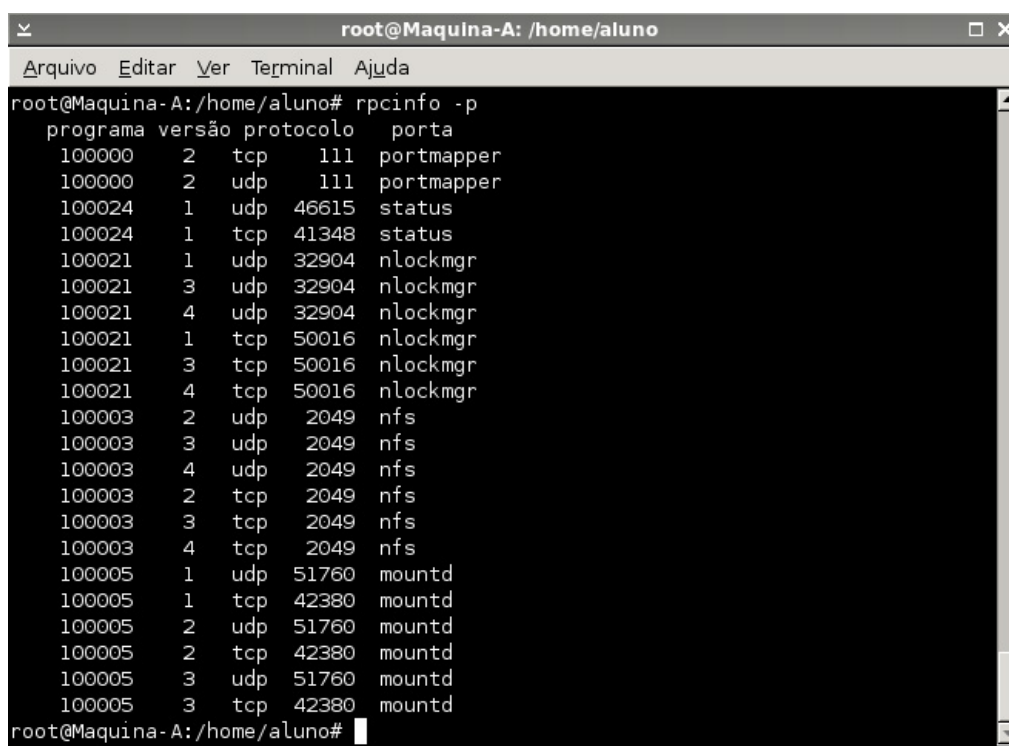
Quando explicamos o portmap, dissemos que o NFS obtém uma porta e a informa ao portmap; e que os clientes perguntam ao portmap pela porta que o NFS está utilizando. Tudo aquilo está correto, mas temos duas coisas a acrescentar.

A primeira é que esse modo de funcionamento utilizando o portmap acontece porque o NFS usa um protocolo chamado RPC (***R**emote **P**rocedure **C**all*) para trocar suas mensagens. Qualquer programa que use RPC utiliza esse esquema do portmap.

A segunda coisa a acrescentar é que embora tenhamos nos referido ao NFS como se fosse um único programa, na verdade o NFS é composto por vários programas, e todos usam RPC.

Existe, por exemplo, um programa responsável por tratar as requisições de montagem, chamado *mountd*, outro responsável por realmente atender as operações que acessam os arquivos e pastas, chamado *nfs*, e um terceiro, chamado *nlockmgr*, encarregado de fazer o *lock* dos arquivos. Os programas que compõem a implementação do NFS no servidor são mostrados na Figura 10. Veja que essa listagem foi obtida com o comando *rpcinfo*.

Figura 10 - Lista de programas que compõem o NFS.



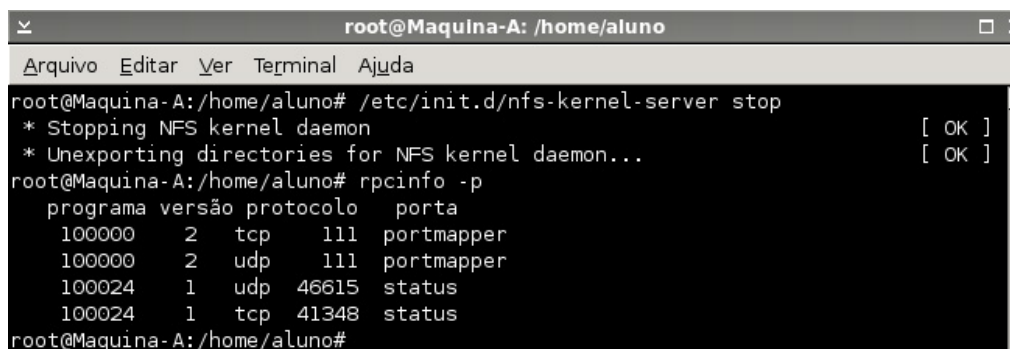
```
root@Maquina-A: /home/aluno# rpcinfo -p
programa versão protocolo  porta
100000      2      tcp      111    portmapper
100000      2      udp      111    portmapper
100024      1      udp     46615  status
100024      1      tcp     41348  status
100021      1      udp     32904  nlockmgr
100021      3      udp     32904  nlockmgr
100021      4      udp     32904  nlockmgr
100021      1      tcp     50016  nlockmgr
100021      3      tcp     50016  nlockmgr
100021      4      tcp     50016  nlockmgr
100003      2      udp      2049  nfs
100003      3      udp      2049  nfs
100003      4      udp      2049  nfs
100003      2      tcp      2049  nfs
100003      3      tcp      2049  nfs
100003      4      tcp      2049  nfs
100005      1      udp     51760  mountd
100005      1      tcp     42380  mountd
100005      2      udp     51760  mountd
100005      2      tcp     42380  mountd
100005      3      udp     51760  mountd
100005      3      tcp     42380  mountd
root@Maquina-A: /home/aluno#
```

Vamos agora parar o serviço NFS e repetir o comando acima. Para parar o NFS, basta digitar o comando a seguir.

```
/etc/init.d/nfs-kernel-server stop
```

O resultado é mostrado na Figura 11. Veja que os programas *nfs*, *mountd*, *nlockmgr* não estão mais executando.

Figura 11 - Programas RPC executando após a parada do NFS.



```
root@Maquina-A: /home/aluno
Arquivo Editar Ver Terminal Ajuda
root@Maquina-A:/home/aluno# /etc/init.d/nfs-kernel-server stop
* Stopping NFS kernel daemon [ OK ]
* Unexporting directories for NFS kernel daemon... [ OK ]
root@Maquina-A:/home/aluno# rpcinfo -p
      programa versão protocolo  porta
      100000    2      tcp      111  portmapper
      100000    2      udp      111  portmapper
      100024    1      udp    46615  status
      100024    1      tcp    41348  status
root@Maquina-A:/home/aluno#
```

Para iniciar novamente o NFS, basta digitar o comando a seguir.

```
/etc/init.d/nfs-kernel-server start
```

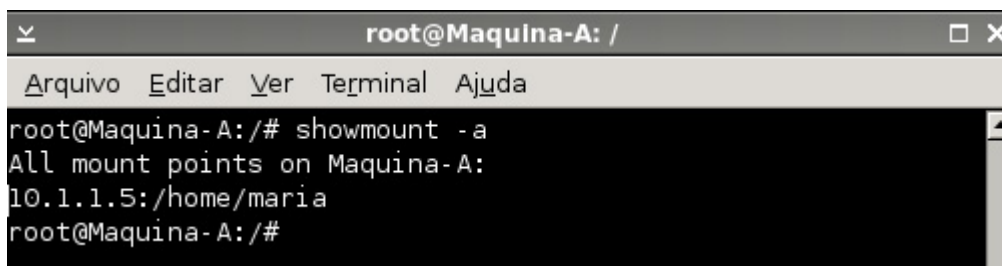
Quando detectar algum problema no NFS, primeiro certifique-se que todos os programas estão executando, conforme mostrado na Figura 10. Caso detecte algum problema, você pode parar e reiniciar o serviço NFS, e o portmap, com os comandos mostrados anteriormente.

Lembre-se que os programas mostrados na Figura 11 foram os programas executando na máquina servidora NFS. Na máquina cliente, o único programa do NFS executando é o *nlockmgr*. Os programas *portmap* e *status* não são propriamente do NFS! Eles são programas para suportar o RPC, portanto sempre devem estar executando.

Cientes Acessando o Servidor

Para verificar quais máquinas clientes montaram quais pastas do servidor, você pode utilizar o comando *showmount*, conforme mostrado na Figura 12. Veja que a máquina cliente 10.1.1.5 montou a pasta */home/Maria*.

Figura 12 - Pastas sendo acessadas pelas máquinas clientes.



```
root@Maquina-A: /
Arquivo  Editar  Ver  Terminal  Ajuda
root@Maquina-A:/# showmount -a
All mount points on Maquina-A:
10.1.1.5:/home/maria
root@Maquina-A:/#
```

Verificando se uma montagem foi bem sucedida

O cliente pode verificar se ele conseguiu montar uma pasta do servidor simplesmente tentando acessar a pasta. Mas se ele quiser, pode digitar o comando `mount`, sem nenhum parâmetro, e ver se aparece uma linha relativa à pasta do servidor. Para o nosso exemplo, essa linha seria como a mostrada a seguir. Veja que são mostrados todos os parâmetros usados no arquivo *fstab*.

10.1.1.1:/home/maria	on	/home/maria	type	nfs
(rw,rsize=8192,wsiz=8192,timeo=14,intr,addr=10.1.1.1)				

Veja aqui a explicação em vídeo sobre o NFS.



Video 06 - NFS



Video 07 - NFS

Atividade 05

1. Qual a função do programa *mountd* que executa no servidor?
2. Qual a função do programa *nfs* que executa no servidor?
3. Qual o nome do comando que é executado no cliente para fazer a associação de uma pasta do cliente com uma pasta do servidor?

Leitura Complementar

Deixamos aqui algumas sugestões de leituras para enriquecer um pouco a aula e para que você possa ler alguns assuntos com mais detalhes, pois não temos como abordar profundamente todo o conteúdo em uma única aula.

- Site da comunidade que desenvolveu o *software* OpenLDAP.
<<http://www.openldap.org/>>
- Artigo mostrando como instalar e utilizar o LDAP.
<<http://www.vivaolinux.com.br/artigo/Entendendo-o-LDAP/>>

Resumo

Nesta aula, você aprendeu a realizar umas das tarefas mais importantes em qualquer rede, que é criar um serviço de autenticação de usuários e compartilhamento de arquivos. Você viu que isso permite que os usuários utilizem qualquer máquina da rede e, ainda assim, seus arquivos vão estar disponíveis naquela máquina. Para implementar o serviço de autenticação, nós utilizamos um servidor LDAP; e para realizar o compartilhamento das pastas, nós usamos o NFS. Nesta aula, utilizamos apenas máquinas Linux, tanto nos clientes quanto no servidor.

Autoavaliação

Assuma que existem duas máquinas em uma rede, chamadas máquina-A e máquina-B, que possuem, respectivamente, os endereços IP 10.1.1.2 e 10.1.1.5. A máquina-A deve funcionar como servidor de arquivos (usando NFS) e autenticação de usuários (usando LDAP), e a máquina-B será o cliente. Responda as seguintes perguntas:

1. Quando se utiliza o LDAP para autenticação é obrigatório o uso do pacote `ldapscripts` para a criação dos usuários ou os usuários podem ser criados no LDAP de outra maneira?
2. Supondo que a máquina-B já está autenticando no LDAP da máquina-A, qual comando poderia ser utilizado na máquina-B para obter as informações do usuário *pedro* que existe na máquina-A?
3. Qual linha, e em qual arquivo, precisa ser adicionada na máquina-A para compartilhar (exportar) a pasta `/home/pedro` apenas para a máquina-B com permissões de somente leitura?
4. Qual linha, e em qual arquivo, precisa ser adicionada na máquina-B para montar a pasta compartilhada no item anterior durante a inicialização da máquina?

Referências

KUROSE, J.; ROSS, K. **Redes de computadores e a internet**. 5. ed. São Paulo: Addison Wesley, 2010.

WETHERALL, David; TANENBAUM, Andrew S. **Redes de computadores**. 5. ed. Rio de Janeiro: Editora Pearson, 2011.