

Redes de Computadores II

Aula 10 - Transfer ncia e Sincroniza o de Arquivos

Apresentação

Nesta aula, iremos aprender a trabalhar com ferramentas que nos permitem transferir arquivos entre computadores remotos através do terminal. Outra importante funcionalidade que aprenderemos nessa aula é como manter a sincronia entre arquivos armazenados em sistemas de arquivos diferentes (como por exemplo em duas máquinas diferentes).

Em aulas anteriores você aprendeu que, utilizando o protocolo SSH (*Secure Shell*), é possível utilizar uma máquina remotamente, ou seja, você pode estar sentado na frente de um computador mas querer utilizar (executar programas, acessar arquivos, etc...) um outro. Porém não aprendemos como transferir, por exemplo, um arquivo da máquina cliente para a máquina servidora ou vice-versa. Isto por que a aplicação que utilizamos, o *ssh*, não provê essa funcionalidade.

Objetivos

Ao final desta aula, você será capaz de:

- Utilizar o SCP para transferência remota de arquivos;
- Utilizar o SFTP para transferência remota interativa de arquivos;
- Utilizar o rsync para transferência local e remota de arquivos;
- Utilizar o rsync para manter arquivos em sistemas de arquivos distintos sincronizados.

O protocolo SCP

No Linux, aprendemos que a aplicação mais utilizada para permitir a utilização de terminais remotos é o **ssh** (*Secure Shell* - Shell Seguro). Naturalmente, existe o programa *ssh* servidor, que deve estar instalado e executando na máquina que você deseja conectar (que chamaremos de servidor), e o *ssh* cliente, que é o programa que você vai utilizar localmente para conectar no computador remoto (servidor). Basicamente, do ponto de vista operacional, o SSH cria um túnel encriptado (seguro) entre o dispositivo cliente e o servidor.

O SCP permite a transferência segura de arquivos entre uma máquina local e um máquina remota. O *scp* utiliza o *ssh* para se conectar a máquina remota, e então copia arquivos para a máquina local, ou vice-versa (copia arquivos da máquina local para a máquina remota).

Um exemplo da utilização do SCP pode ser vista abaixo:

```
1 [aluno@imd] $ scp exemplo.pdf danilo@10.7.34.77:/home/aluno/  
2 aluno@10.7.34.77's password:  
3 residencia.pdf          100% 395KB 394.7KB/s  00:00  
4 [aluno@imd] $
```

Utilizando o scp

A utilização do comando *scp* é análoga a utilização do comando de cópia de arquivos locais no Linux, o comando *cp* (*copy*). A diferença é que, com o *scp*, você está copiando arquivos entre diferentes hosts, ao invés de estar copiando arquivos somente de um local para outro dentro do sistema de arquivos do seu computador.

A utilização normal do comando *cp* no terminal de um sistema Linux se dá da seguinte maneira:

```
1 cp /home/imd/helloworld.java /home/imd/tecnico
```

Este comando copiará o arquivo *helloworld.java* do diretório `/home/imd/` para o diretório `/home/imd/tecnico/`.

Agora vamos dizer que eu esteja no meu computador, em casa, e precise copiar um arquivo local chamado *helloworld.java* para o meu *laptop* do trabalho (máquina remota). No computador da minha residência, o caminho de diretórios (localização) para o meu arquivo *helloworld.java* é `/tmp/helloworld.java`. No meu *laptop*, o diretório destino que desejo que o arquivo seja armazenado é `/home/imd/tecnico/`; Porém, como este diretório está no computador remoto, eu preciso adicionar um prefixo com as minha credenciais (nome de usuário) para autenticação na máquina remota.

Então, para enviar um arquivo da máquina local para um máquina remota, o seguinte comando pode ser utilizado no terminal (para o exemplo descrito anteriormente, considerando que o nome de usuário é *aluno* e o endereço IP da meu *laptop* remoto é 10.1.1.1):

```
1 scp /tmp/helloworld.java aluno@10.1.1.1:/home/imd/tecnico
```

Note que este prefixo é familiar: é o mesmo argumento que você utilizou no comando *ssh* em aulas passadas para se conectar a uma máquina remota. Essa é a mesma autenticação utilizada para conexão via SSH, haja vista que o como já foi dito anteriormente, o *scp* utiliza o *ssh* para se conectar ao dispositivo remoto. Note que existe um caractere dois pontos (:) entre o prefixo e o caminho de diretórios. Você poderá então ser solicitado a entrar com a senha do usuário *aluno* na máquina 10.1.1.1. Devidamente autenticado, a cópia do arquivo será iniciada, permitindo que o usuário verifique o progresso da transferência de todos arquivos (neste caso é somente um arquivo) até o seu término.

Agora você sabe transferir arquivos do seu computador residencial para o seu *laptop* que se encontra em outro lugar. Para fazer uma cópia inversa (do *laptop* remoto para o computador local), basta inverter a ordem dos dois arquivos:

```
1 scp aluno@10.1.1.1:/home/imd/tecnico/helloworld.java /tmp
```

Um exemplo mais útil pode ser a transferência de um diretório completo, do seu computador para o *laptop*. Suponha que temos um diretório local com diversos arquivos e subdiretórios dentro (`/home/imd/tecnico/2017/dados`) e queria

copiar para o computador remoto dentro do diretório `/tmp/redes`. Para copiar o diretório `dados` e todos os arquivos e sub-diretórios contidos neles, usamos a *flag* `-r` (recursivo) junto ao comando `scp`:

```
1 scp -r /home/imd/tecnico/2017/dados aluno@10.1.1.1:/tmp/redes
```

Note que não foi incluído o nome do diretório `dados` no caminho do computador remoto. O comando `scp` irá criar uma cópia do diretório `dados` dentro de `/tmp/redes` (esse caminho já deverá existir!). Caso o caminho de diretórios `/tmp/redes` já existisse no computador remoto, todos os arquivos dentro deles seriam sobrescritos.

Atividade 01

1. Escreva um comando, utilizando o `scp`, para transferir da sua máquina local o arquivo `/home/aluno/teste.txt` para o diretório `/tmp` de uma máquina remota endereçada pelo IP 10.8.0.2. Considere que o nome do usuário na máquina remota é "aluno".
2. Escreva um comando, utilizando o `scp`, para transferir para o diretório `/home/aluno` a sua máquina local o arquivo `/var/local/lista.txt` presente em uma máquina remota endereçada pelo IP 10.8.0.9. Considere que o nome do usuário na máquina remota é "aluno".

Para verificar as respostas [clique aqui](#).

Respostas

1. `scp /home/aluno/teste.txt aluno@10.8.0.2:/tmp`
2. `scp aluno@10.8.0.9:/var/local/lista.txt /home/aluno`

O protocolo SFTP

Outra maneira de transferir arquivos entre computadores distintos porém com conectividade entre eles é um protocolo conhecido como SFTP (*SSH File Transfer Protocol*). O utilitário *sftp*, presente em ambientes Linux, implementa esse protocolo e permite a transferência interativa de arquivos entre dois sistemas computacionais.

Um exemplo da utilização do SFTP pode ser vista abaixo:

```
1 [aluno@imd] $ sftp danilo@10.7.34.75
2 aluno@10.7.34.77's password:
3 Connected to 10.7.34.77.
4 sftp> ls
5 residencia.pdf
6 sftp> ll
7 rg_cpf.pdf
8 sftp> put rg_cpf.pdf /home/aluno/
9 Uploading rg_cpf.pdf to /home/aluno/rg_cpf.pdf
10 rg_cpf.pdf          100% 531KB 531.2KB/s  00:00
11 sftp> exit
12 [aluno@imd] $
```

Utilizando o sftp

Para utilizar o *sftp*, na linha de comando (terminal) digite o seguinte comando:

```
1 sftp aluno@10.1.1.1
```

Onde **aluno** é o nome de usuário na máquina endereçada pelo IP 10.1.1.1. Se solicitado, a senha referente ao nome de usuário deverá ser inserida. Assim que estiver conectado, o utilitário *sftp* irá indicar o seguinte terminal, onde comandos podem ser inseridos:

```
1 sftp>
```

Devidamente conectado, a maioria dos comandos usados para navegar em ambientes Linux são funcionais dentro do ambiente oferecido pelo *sftp*, permitindo assim navegar e manipular diretórios e arquivos no computador remoto (cd, pwd, etc.). Para navegar e manipular diretórios e arquivos no computador local, basta acrescentar a letra l (L minúsculo) no início dos comandos (lcd, lpwd, etc.).

Se precisar transferir arquivos do computador local para o computador remoto, o seguinte comando pode ser utilizado:

```
1 sftp> put <arquivo_local> <destino_no_computador_remoto>
```

Então, se por acaso quisermos transferir o arquivo helloworld.java do disco local para o diretório /tmp na máquina remota, o seguinte comando deverá ser utilizado:

```
1 sftp> put helloworld.java /tmp
```

Caso quisermos transferir um arquivo remoto para o computador local, o seguinte comando deverá ser utilizado:

```
1 sftp> get <arquivo_remoto> <destino_no_computador_local>
```

Então, se por acaso quisermos transferir o arquivo /tmp/helloworld.java do disco remoto para o diretório /home/imd/ na máquina local, o seguinte comando deverá ser utilizado:

```
1 sftp> get /tmp/helloworld.java /home/imd
```

Atividade 02

1. Escreva um comando, na sintaxe do SFTP, para transferir o arquivo remoto /home/aluno/teste.txt da máquina 10.7.6.5 para o diretório /tmp/ da sua máquina local.

Para verificar a resposta [clique aqui](#).

Respostas

1. `get /home/aluno/teste.txt /tmp`

O utilitário *rsync*

O *rsync* é um utilitário que tem como objetivo otimizar ao máximo a transferência de arquivos entre um computador remoto e outro computador local. Essa otimização é realizada a partir de uma premissa da sua implementação: durante a transferência de arquivos, somente são transmitidas as diferenças entre os arquivos nas duas máquinas, evitando assim a transferência de arquivos que não foram alterados ou que já existem. Caso o arquivo não exista, ele é enviado na sua forma integral. Caso tenha sido modificado, somente os dados alterados são transferidos.

O *rsync*, assim como o *scp*, pode utilizar o *ssh*, provando assim uma conexão segura, criptografada e autenticada entre as partes.

Por esses motivos, cada vez mais o *rsync* é utilizado para a realização de espelhamento de arquivos e para *backup* de dados. Muitos conhecidos repositórios de arquivos que mantêm *mirrors* (servidores secundários redundantes) espalhados em máquinas remotas fazem uso do *rsync* para manter a sincronização entre eles.

Um exemplo da utilização do RSYNC pode ser vista abaixo:

```
1 [aluno@imd] $ rsync -avz prova.pdf danilo@10.7.34.77:/home/aluno
2 aluno@10.7.34.77's password:
3 building file list ... done
4 prova.pdf
5
6 sent 106629 bytes received 42 bytes 42668.40 bytes/sec
7 total size is 115139 speedup is 1.08
8 [aluno@imd] $
```

Utilizando o *rsync*

A utilização do utilitário *rsync* segue o seguinte formato:

```
1 rsync [opções] [caminho_origem] [caminho_destino]
```

Onde pelo menos um dos caminhos deve ser local. O outro caminho pode ser local (para transferência em uma mesma máquina) ou remota (para transferência através da rede). Para indicação de um caminho remoto, o campo [host] deve ser incluído e separado por dois pontos (:), como nos formatos a seguir:

```
1 rsync [opções] [host_remoto]:[caminho_origem] [caminho_destino]
2 rsync [opções] [caminho_origem] [host_remoto]:[caminho_destino]
```

Copiando arquivos localmente

O *rsync* pode ser utilizado mesmo para transferências locais, se tornando interessante por não transferir arquivos que já estão no diretório de destino. O exemplo abaixo copia recursivamente (opção *-a*) todo o conteúdo do diretório `/home/imd/aulas/` para dentro do diretório `/usr/local/backup/` (a opção *-v* indica detalhes da transferência no terminal):

```
1 rsync -av /home/imd/aulas/ /usr/local/backup/
```

Copiando arquivos remotamente

Para transferências remotas, vamos manter o mesmo exemplo anterior, porém agora os arquivos devem ser transferidos para o diretório `/usr/local/backup/` de um servidor remoto endereçado por `10.1.1.1` (considere um nome de usuário "aluno" na máquina remota).

```
1 rsync -avz /home/imd/aulas/ aluno@10.1.1.1:/usr/local/backup/
```

Neste caso o *rsync* irá fazer uso do *ssh* para transferência dos arquivos para a máquina remota. Desta forma, ele poderá solicitar autenticação (senha do usuário na máquina remota). Note que no exemplo anterior a opção *-z* foi adicionada, pois desta forma os dados transferidos são compactados gerando um economia de banda.

Sincronizando dois diretórios

Considere que você chegou no trabalho e deseja sincronizar o diretório `/home/imd`, da sua máquina de casa (identificada pelo endereço `teste.fulano.com.br`), com o diretório `/tmp/imd` no seu notebook do trabalho, tornando assim o conteúdo do último igual ao do primeiro (considere um nome de usuário "aluno" na máquina remota). Considere o seguinte comando:

```
1 | rsync -avz --delete aluno@teste.fulano.com.br:/home/imd/ /tmp/imd/
```

Note a presença de uma nova opção `--delete`, que apaga do destino qualquer arquivo que não esteja na origem, tornando assim o conteúdo do destino exatamente igual o da origem. Um detalhe importante vale a pena ser destacado, quando o diretório é especificado com a barra (`/`) no final do caminho, somente o conteúdo do diretório é copiado. Quando é usado sem a barra, o diretório é copiado. Ou seja, neste último exemplo somente o conteúdo do diretório `/home/imd/` da máquina remota será sincronizado, e não a pasta `imd` propriamente dita.

Atividade 03

1. Escreva um comando, utilizando o *rsync*, para transferir da sua máquina local o diretório `/home/aluno/exercicio` e seu conteúdo para o diretório `/tmp` de uma máquina remota endereçada pelo IP `10.8.0.2` (considere um nome de usuário "aluno na máquina remota").

Para verificar a resposta [clique aqui](#).

Respostas

1. `rsync -avz /home/aluno/exercicio aluno@10.8.0.2:/tmp/`

Qual serviço de transferência de arquivos utilizar?

O uso de qual tipo de utilitário vai depender do contexto da sua utilização. O `rsync` foi concebido para manter dois locais em sincronia. Você pode modificar vários arquivos de um lado e solicitar ao `rsync` para espelhar essas modificações para o outro lado. `Rsync` analisa somente as partes modificadas para transferir, o que o faz mais eficiente quando:

- Você não tem uma lista dos arquivos que foram modificados;
- Você precisa que os arquivos nos dois lados tenham a mesma permissões, mesmas datas de modificação, mesmos donos (*owners*), etc.;
- O número total de arquivos a serem transferidos é grande (ou seja, muitos arquivos);
- A quantidade de arquivos modificados é consideravelmente menor que o tamanho total da transferência (ou seja, transferência de grandes arquivos).

Caso a transferência não esteja incluídas nos pontos destacados, um simples comando `scp` ou `sftp` pode desempenhar esse papel de maneira mais eficiente e rápida.

Resumo

Nesta aula estudamos diferentes maneiras de se realizar a transferência de arquivos e diretórios em sistemas Linux. Desde comandos mais simples, como o *scp*, passando por ferramentas que permitem interatividade ao usuário, como o *sftp* até utilitários mais completos e versáteis como o *rsync*. Ainda assim, todas essas maneiras permitem a transferência de maneira segura, através do protocolo SSH.

O conhecimento destas ferramentas, mesmo que superficialmente, é essencial para profissionais da área de informática que serão responsáveis por administrar ambientes que utilizam infraestruturas com sistemas operacionais Linux.

Autoavaliação

1. Utilizando máquinas virtuais em ambiente Linux, pratique a transferência de arquivos entre *hosts* remotos utilizando:
 1. *scp*
 2. *sftp*
 3. *rsync*

Referências

Nemeth, Snyder, Heyn, Whaley: UNIX and Linux System Administration Handbook

Kernighan & Pike: The UNIX programming environment

Stevens & Rago: Advanced Programming in the UNIX Environment