UFF - Universidade Federal Fluminense
Escola de Engenharia
Curso de Engenharia de Telecomunicações
Programa de Educação Tutorial
Grupo PET-Tele

Manual de Acesso
a
Servidores SSH

Autor : Vinicius Puga de Almeida Santos
Orientador: Alexandre Santos de la Vega (Tutor do grupo PET-Tele)

Setembro/2009

Capítulo 1

Introdução

O SSH é extremamente prático para se usar aplicativos que estão instalados em seu computador, e fazer o gerenciamento remoto de um servidor e dos serviços que este presta.

No entanto, para utiliza-lo na obtenção ou envio de arquivos, a linha de comando torna-se muitas vezes pouco prática, especialmente quando se deseja receber ou enviar muitos arquivos. Para isso escrevi este tutorial ensinando a utilizar o SSH para enviar ou receber arquivos de uma forma extremamente facil, tentando eliminar as dificuldades que muitos possuem ao se utilizar deste recurso.

Esperamos que seja útil!

Vinicius Puga e Grupo PET-Tele

Capítulo 2

Acesso ao servidor SSH por diferentes plataformas

2.1 Primeiro Passo: Configurando o servidor

A instalação de um servidor SSH em um sistema "Debian" (ou semelhante a este, como o Ubuntu) é simples. Basta executar o comando:

Comando	Função
sudo apt-get install ssh	Faz a instalação de um servidor SSH e o habilita com as configurações padrão

A configuração de um servidor SSH envolve a edição de seu arquivo de configuração que é localizado em :

Localização	Função
/etc/ssh/sshd_config	Arquivo de configuração global de um servidor SSH

Neste trabalho abordaremos apenas os aspectos de configuração do cliente, por ser destinado aos usuários do serviço.

2.2 Gerando uma chave criptográfica

Alguns servidores SSH utilizam para a autenticação do usuário um arquivo contendo uma chave criptográfica que, espera-se, que apenas o usuário real irá possuir. Neste momento, sempre surge a pergunta: "Porque não utilizar apenas uma senha?". A resposta para esta pergunta está no fato de senhas serem mais vulneraveis do que chaves criptográficas. Senhas podem ser submetidas a "ataques de dicionário", ou capturadas por determinados programas se outros computadores forem usados no acesso ao servidor.

Tomando-se os devidos cuidados, como por exemplo mantendo-se o arquivo em uma mídia separada (disquete, CD ou DVD, pendrive) e longe da internet, o nível de segurança alcançado com a chave criptográfica é muito superior ao de uma senha.

O comando abaixo pode ser usado para criar uma chave criptográfica com o algoritmo RSA de tamanho igual a 1024 bits e deve ser executado no servidor que será acessado - embora seja possível executar o comando no computador local e enviar a chave ao servidor colocando-a em seu "path" (caminho) padrão . É recomendável manter as opções padrão, apenas pressionando

ENTER para todas as opções.

Comando	Função
ssh-keygen -t rsa -b 1024	Gera uma chave criptográfica

As chaves criadas serão gravadas nos arquivos */home/usuario/.ssh/id_rsa* e */home/usuario/.ssh/id_rsa.pub* (sendo respectivamente suas chaves privada e públicas). É muito importante que o acesso a sua chave tenha acesso restrito, isto pode ser feito alterando as permissões de acesso para 750, com o comando abaixo:

Comando	Função
chmod 750 /.ssh	Restringe as permissões de acesso da pasta de SSH

Em seguida, devemos copiar a chave pública para *∼/.ssh/authorized_keys* . Este procedimento habilita sua chave recém-criada como uma chave autorizada.

Comando	Função
cp ∼/.ssh/id_rsa.pub ∼/.ssh/authorized_keys	Copia a chave para o caminho padrão

Isto conclui a parte respectiva a criação da sua chave. Mantenha ela em um local seguro e jamais transmita-a pela internet. A seguir, é descrito como o acesso deve ser feito (utilizando-se dela, ou não).

2.3 Acessando pelo Sistema Operacional Linux

2.3.1 Método 1 – Usando o ambiente GNOME

1. Abra uma janela qualquer do gerenciador de arquivos
2. No campo Localização digite *ssh://usuario@algum.servidor.com* e pressione ENTER
3. Caso seja pedida uma senha, digite a senha do seu usuário e pressione ENTER.
4. Os arquivos mostrados na janela estão no servidor. As operações de mover e copiar arquivos podem ser feitas através daquela janela

OBS.: Caso o servidor a ser acessado necessite de uma chave criptográfica para a autenticação, a mesma deverá ser colocada na pasta */home/usuario/.ssh/* com o nome de "id_rsa" no computador que está sendo usado.

2.3.2 Método 2 – Usando o ambiente KDE

1. Abra uma janela qualquer do gerenciador de arquivos
2. No campo Localização digite *fish://usuario@algum.servidor.com* e pressione ENTER
3. Caso seja pedida uma senha, digite a senha do seu usuário e pressione ENTER.
4. Os arquivos mostrados na janela estão no servidor. As operações de mover e copiar arquivos podem ser feitas através daquela janela

2.3.3 Método 3 – Usando a linha de comando

Estando com um terminal aberto, utilize as seguintes linhas de comando de acessar o servidor desejado:

Usando o cliente de SSH

Este comando deve ser usado para acessar o computador remoto como um usuário "local". É extremamente útil para executar comandos, controlar aplicativos (com interface texto) , iniciar ou parar serviços, etc.

Comando	Função
ssh usuario@algum.servidor.com	Faz o login no servidor e disponibiliza uma linha de comando.

Usando o cliente de SFTP

Este comando deve ser usado preferencialmente acessar o computador remoto (servidor) via SSH como seria feito com um servidor de FTP. Sua recomendação é para a transferência (envio e recebimento) de arquivos.

Comando	Função
sftp usuario@algum.servidor.com	Acessa o servidor e disponibiliza um terminal para os comandos abaixo.
lcd	Muda de diretório no computador local
lls	Lista os arquivos no computador atual
cd	Muda de diretório no computador remoto (servidor)
ls	Lista os arquivos da pasta atual no computador remoto (servidor)
pwd	Mostra o diretório atual no computador remoto (servidor)
lpwd	Mostra o diretório atual no computador local
put xxx.xx	Envia o arquivo "xxx.xxx" para a pasta atual no computador remoto
get xxx.xx	Recebe o arquivo "xxx.xxx" para a pasta atual no computador local
exit	Termina a sessão

Usando o cliente de SCP

É uma ferramenta semelhante ao SFTP porém mais primitiva ao permitir apenas a transferencia de arquivos entre o servidor e o computador local.Seu uso é recomendado na criação de *scripts* para transferência automatizada de arquivos devido a ter uma interface mais automatizada.

Comando	Função
scp arquivo.local usuario@servidor:/destino/	Transmite arquivo.local do cliente para o servidor em "/destino/"

2.4 Acesso pelo Sistema Operacional Microsoft Windows

O serviço SSH não é exclusividade de plataformas derivadas de UNIX, podendo também ser acessado por máquinas contendo o *Microsoft Windows*. Para isto, é necessário o download de softwares específicos, visto que o sistema em si não possui suporte a este serviço nativamente.Ressaltam-se também que existem certas incompatibilidades entre as chaves criptográficas usadas em programas diferentes. Portanto estas devem ser convertidas para um formato que o cliente SSH que será utilizado no Windows seja capaz de compreender.

2.4.1 Conversão da chave criptográfica

A utilização do serviço de SSH no Windows requer que a chave criptográfica obtida em um servidor Linux/Unix (criada pelo *OpenSSH* ou *SSH* proprietário) seja convertida para um formato que possa ser utilizado neste sistema operacional. O procedimento abaixo descreve como esta conversão pode ser feita:

1. Obtenha o programa de conversão em :
   http://tartarus.org/∼simon/putty-snapshots/x86/puttygen.exe

2. Abra o *puttygen.exe* e clique em **Conversions ▷ Import Key** e selecione sua chave.

3. O programa irá importá-la, o proximo passo é converte-la e salvá-la no formato-padrão para Windows. Para isso clique em **File ▷ Save Private key** (não se esqueca de salvar com um nome diferente).

4. A chave está convertida para o formato PPK, usado pelo Cliente PuTTY e derivados. O próximo passo é a escolha do cliente a ser usado.

2.4.2 Acesso pelo cliente PuTTY

O PuTTY é o cliente mais antigo para acesso a servidores SSH usando o Windows. Ele fornece ao usuário um ambiente muito semelhante ao encontrado em ambientes *NIX (UNIX ou LINUX), provendo ao mesmo uma linha de comando, cuja manipulação de arquivos e serviços deve ser feita conforme o cliente padrão dos sistemas *NIX.

Procedimento de instalação:

1. Obtenha o programa em:
   http://the.earth.li/ sgtatham/putty/latest/x86/putty.exe

2. Instale-o e o abra.

3. Ao abrir o programa, selecione Session no lado esquerdo. No lado direito entre com o endereço do servidor a ser acessado em *Hostname* e escolha em *Protocol* a opção *SSH*.

4. No lado esquerdo, selecione *Connection ▷ SSH ▷ Auth*, e no lado direito no campo *Private Key* clique em *Browse* e localize a sua chave criptográfica no formato PPK

5. Clique no botão na parte de baixo *Open*

6. O PuTTY tentará estabelecer uma conexão com o servidor e em seguida abrirá uma linha de comando, onde podem ser usados os mesmos comandos descritos acima para o cliente linux

2.4.3 Acesso pelo cliente WinSCP

O Cliente WinSCP tem uma interface semelhante ao de um programa de FTP, onde à esquerda se localizam os arquivos locais e a direita os arquivos do servidor, facilitando bastante a manipulação de arquivos entre o servidor e o computador local. Em contrapartida este cliente não oferece uma linha de comando o que limita as operações no servidor a apenas a manipulação e edição de arquivos.

Procedimento de Instalação:

1. Obtenha o programa de instalação em
   http://winscp.net/eng/download.php

2. Ao término da instalaçao, abra o programa. Será apresentado um diálogo de configuração, que deve ser preenchido conforme as opções abaixo:
   Host Name: *Endereço_do_servidor*
   Port number: 22
   Username: *usuario_remoto*
   Password: *senha,se houver*
   Private Key File: Caminho para a chave criptográfica PPK
   File Protocol: *SFTP*

Capítulo 3

Tunelamento via SSH

Uma das mais úteis características do SSH é a sua capacidade de criar túneis criptográficos. Em se tratando de um ramo do protocolo TCP/IP, o SSH pode criar uma conexão entre dois computadores, intermediada por um servidor remoto, fornecendo a capacidade de redirecionar pacotes de dados. Esta técnica permite, entre outras coisas, navegar na internet com segurança por meio de uma conexão insegura (como uma rede wireless sem criptografia), ou contornar as restrições de serviços impostas por um firewall em uma rede local.

O tunelamento pelo serviço SSH aloca uma porta no computador local, de forma que todas as informações trafegadas por esta porta serão redirecionadas para o servidor remoto que fará a conexão com a rede externa (como a Internet). Os aplicativos que desejarem utilizar a conexão túnel deverão ser configurados para utilizarem um *Proxy SOCKS*, com o endereço *"127.0.0.1"* e cuja porta deverá ser a mesma escolhida durante criação do túnel.

3.1 Tunelamento no Windows com o PuTTY

O cliente PuTTY é capaz de fazer o tunelamento alterando-se certas opções em sua configuração.

Procedimento de Configuração:

1. Na janela de configuração do PuTTY, entre com as opções de conexão mostradas anteriormente neste tutorial para a configuração básica do PuTTY
2. No lado esquerdo, selecione *Connection* ▷ *SSH* ▷ *Tunnels*, e no lado direito marque a opção *"Local ports accept connections from other hosts"*
3. Ainda no lado direito, em "Add new forwarded port", preencha o campo "Source Port" com um número maior do que 1000 (neste exemplo assumiremos a escolha da porta aleatória 1080)
4. Logo abaixo entre com o endereço do servidor SSH ao qual será feita a conexão, e marque abaixo as opções *Dynamic* e *Auto*
5. Pressione Open. O túnel está estabelecido!

3.2 Tunelamento no Linux usando o OpenSSH

O tunelamento no Linux é mais simples do que em outros sistemas operacionais. É apenas necessário executar o comando abaixo no shell:

Comando	Função
ssh -D 1080 usuario@algum.servidor.com	Cria um túnel SSH redirecionado pela porta 1080.

OBS.: Qualquer porta acima de 1000 pode ser especificada e não apenas a porta 1080.

3.3 Forma de uso

As aplicações que desejarem utilizar o serviço (como browsers, clientes de email, mensagens instantâneas e etc) devem ser configurados para utilizarem uma conexão proxy como especificado abaixo:

Tipo de Proxy:	SOCKS
Endereço do Proxy:	127.0.0.1
Porta:	1080 (ou outra que tenha sido especificada na criação do túnel)
Versão do SOCKS:	SOCKSv5