

SSH

set 03, 2021

Rafael H.

5min de leitura

Lista de Comandos SSH



Este é um tutorial que irá abordar os **Top 13** Comandos Básicos do [SSH](#). Estes comandos irão lhe dar um entendimento básico de como navegar e trabalhar em um terminal com arquivos [Linux](#).

- ls** Mostra o conteúdo do diretório (lista de nome dos arquivos).
- cd** Mudar Diretório.
- mkdir** Criar uma nova pasta (diretório).
- touch** Criar novo arquivo.
- rm** Remover um arquivo.
- cat** Mostrar conteúdo de um arquivo.
- pwd** Mostra diretório atual (Caminho completo de onde você está agora).
- cp** Copiar arquivo/pasta.
- mv** Mover arquivo/pasta.
- grep** Buscar por frase especifica em arquivo/linha.
- find** Buscar arquivo ou diretório
- vi/nano** Editor de Texto.
- history** Mostrar últimos 50 comandos usados.

Conteúdo

- O que você vai precisar?
- Passo 1 — Acessar um Servidor Remoto
- Passo 2 — Aprender os Comandos SSH
- Conclusão

O que você vai precisar?

Passo 1 — Acessar um Servidor Remoto

Recomenda-se ter um servidor virtual com um modelo recém-construído para que, se você acidentalmente excluir algo que não devia a ser excluído, você pode reconstruir o servidor e começar tudo a partir do zero.

SSH significa **Secure Shell**. É um protocolo usado para se conectar com segurança a um servidor / sistema remoto.

O comando básico que você deve usar é:

ssh user@serverip

Este comando conecta você a um servidor que tem um endereço IP do servidor e nome de usuário. Outra maneira, ainda mais simples de se conectar seria usar ssh do IP do servidor, dessa forma o SSH vai pensar que você está se conectando com o **mesmo** usuario que você está conectado com agora.

Depois de inserir este comando, você precisa colocar uma senha (se você estiver se conectando pela primeira vez, também será pedido com uma mensagem de aviso de que o servidor que você está se conectando não é reconhecido, basta digitar sim na [linha de comando](#)).

É isso, você está conectado e pode continuar lendo este guia sobre como gerenciar seus arquivos via Terminal! Se você quiser sair do servidor remoto e voltar para sua máquina local, basta digitar exit na linha de comando.

Um guia mais detalhado sobre como se conectar ao VPS usando o cliente de SSH do Putty pode ser encontrado [aqui](#).

Passo 2 — Aprender os Comandos SSH

Nesta etapa, iremos passar pelos comandos SSH principal/mais usados que você deve aprender!

IMPORTANTE: Isso se aplica a todos os comandos SSH. Ao escrever um argumento próximo ao comando, por exemplo, cd 'Pasta Um' (onde o nome da pasta consiste em duas palavras separadas), você deve digitar o nome da pasta entre aspas. Comando cd A Pasta Um (sem aspas) não funcionará, pois o SSH a interpretará com dois argumentos ("Pasta" e "Um").

ls – Este comando é usado para listar todos os arquivos e diretórios. Recomendamos usar este comando com uma opção -l, que seria ls -l, de tal forma, todos os arquivos serão listados de uma maneira mais conveniente e com mais detalhes / informações sobre eles. Outra opção útil é -a, isso também mostrará todos os outros arquivos também, que inclui arquivos / diretórios **ocultos** (arquivos de ponto com um . na frente deles, por exemplo: diretório .ssh).

cd – Este comando é usado para “andar” entre diretórios (cd significa do inglês “alterar diretório”). Depois de digitar o comando **cd** e um arquivo e diretórios com ls, você pode escolher um diretório para “andar”. Por exemplo, digite o comando **cd ..** para ir para o diretório anterior. Você também pode digitar um caminho completo para ir para uma home de diretório que você deseja inserir. Digite o comando **cd home** e você será levado instantaneamente ao seu local atual para `home`. Você pode tentar usar ls novamente para ver que a informação que foi enviada para a tela foi alterada. Você também pode digitar um caminho completo para um determinado diretório se, por exemplo, você quiser inserir um diretório que é de poucos níveis de profundidade. Você pode usar por exemplo: `cd home / TestDirectory / AnotherDirectory`. Desta forma, você será imediatamente levado para o diretório chamado “AnotherDirectory”. Use o comando `cd ..` (espaço e dois pontos após cd) para subir um nível (no nosso exemplo, seremos movidos de volta para “TestDirectory” de “AnotherDirectory”).

mkdir – Este comando é usado para criar um novo diretório (significa “make directory”). Ele simplesmente cria um novo diretório com o nome escolhido, por exemplo `mkdir NewFolder` irá criar um novo diretório com um nome “NewFolder” no seu diretório ativo atualmente (onde você está agora).

touch – Este comando é usado para criar um novo arquivo com a extensão escolhida. Por exemplo, toque em `NewFile.txt` irá criar um novo arquivo “txt” chamado “NewFile” no seu diretório atual (a extensão poderia realmente ser qualquer coisa que você quiser, você pode até mesmo criar um novo arquivo sem extensão em tudo, por exemplo toque `NewFile`).

rm – Este comando é usado para remover um arquivo / diretório escolhido. Por exemplo, `rm NewFile` removerá o arquivo criado anteriormente chamado “NewFile”. Se você deseja remover um diretório e todos os diretórios dentro dele, use `rm -r NewFolder`, isso irá excluir a pasta “NewFolder” e todas as outras pastas dentro dela.

cat – Este comando é usado para exibir o conteúdo do arquivo. Por exemplo, `cat info.txt` trará o conteúdo do arquivo para a tela. Outro exemplo: `info.txt do gato info2.txt > mergedinfo.txt` irá mesclar dois arquivos juntos (“info.txt” e “info2.txt”) e escrever o conteúdo mesclado para um arquivo “mergedinfo.txt”.

pwd – Este comando mostra sua localização atual no sistema de arquivos. Por exemplo, digite `pwd`, e pode mostrar algo como isto: “home / user / public_html”.

cp – Este comando é usado para copiar arquivos e pastas. A sintaxe é:

cp [opções] source dest

Basicamente, em vez de source, você escreve o arquivo que você deseja copiar. Em vez de dest, escreva o caminho de destino / pasta / arquivo. Agora, se você escreveu um nome de destino que não existe, por exemplo, você tem um arquivo de origem `oldfile.txt` e escreve um arquivo de destino `newfile.txt`, o bash simplesmente copiará o arquivo e o colará com um novo nome.

Além disso, aqui estão algumas opções que você pode usar com um comando `cp`:

cp -f source dest – Força o procedimento de cópia removendo o arquivo de destino, se necessário.

cp -i source dest – Dar-lhe-á uma mensagem de aviso antes de substituir o arquivo.

cp -u source dest – Atualizar opção. Só copiará se o arquivo de origem for mais novo que o arquivo de destino.

cp -n source dest – Não copiará se o arquivo já existir (não sobrescrever).

cp -a source dest – Esta opção irá arquivar os arquivos.

1. **Mmv** – Este comando funciona da mesma forma que **cp**, mas em vez disso, **move** o arquivo em vez de copiá-lo. Esse comando também pode ser usado para **renomear** o arquivo. Se tomarmos o mesmo exemplo do comando **cp**, (no nosso diretório atual, teremos um arquivo `oldfile.txt`) e escreveremos este comando: `mv oldfile.txt newfile.txt` bash simplesmente renomeará o arquivo `oldfile.txt` para `newfile`. TXT.
2. **grep** – Este comando procura uma string dada em arquivos / pastas. Por exemplo: o arquivo `grep` ‘word’ procuraria uma palavra ‘word’ em um arquivo chamado **“file”**. **Grep** retornará toda a linha do arquivo se a frase for encontrada. Por exemplo, há uma linha ‘Tudo em tudo é apenas outra palavra em uma frase’ no arquivo chamado **“arquivo”**, usando o comando `grep ‘word’ arquivo`, esta linha será enviada para a tela desde que a palavra é encontrada.

1. **find** – Agora este comando é usado para pesquisar pastas para um arquivo ou arquivos que atendam aos critérios dados (nome, tamanho, tipo de arquivo). Por exemplo: `find. -name “*.html”` Este comando irá mostrar todos os arquivos no diretório atual que têm uma extensão / extensão de “.html” (observe como usamos o símbolo “*” no nosso comando, que é um coringa que basicamente diz ao bash que Não importa como o arquivo é nomeado **antes** da extensão “.html”, é apenas importante que o arquivo **termine** com “.html”).

1. **vi / nano** – Este comando é usado para entrar em um editor de texto. Por exemplo, `nano newfile` irá criar um novo arquivo chamado “newfile” e iniciar o **nano** editor ou editar um arquivo existente

“newfile” (se existir) com o nano editor. A mesma coisa se aplica para o comando vi, que inicia outro editor chamado “vi”.

IMPORTANTE: nano, ao contrario vi nao e um editor paarao, voce provavelmente vai ter que instala-lo primeiro se você deseja usá-lo. Você pode encontrar um guia sobre como instalar o editor nano [aqui](#).

history – Este comando é usado para exibir os últimos comandos usados. Por exemplo: o histórico 20 mostrará os últimos 20 comandos inseridos no **Terminal**.

Conclusão

Esperamos que este tutorial tenha sido suficientemente claro para que você possa entender os comandos SSH e como usá-los. Recomendamos que consulte esta [página](#) para obter mais detalhes sobre os comandos explicados neste tutorial e muitos outros! Para mais conteúdo sobre comandos SSH, consulte nossos Tutoriais.

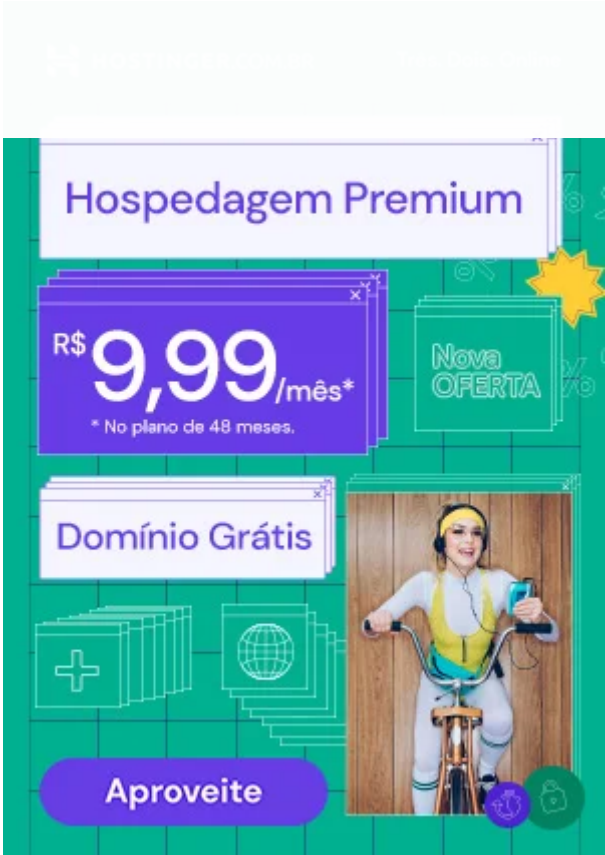
O AUTOR

Rafael H.

Trabalho com marketing digital desde 2017. Atualmente sou Country Manager do Brasil na Hostinger. Sou apaixonado pelo mundo da tecnologia e hospedagem de sites, tendo experiência com WordPress, marketing digital, SEO, copywriting e ferramentas de automação de marketing. Também já trabalhei como analista de conteúdo, redator, assessor de imprensa e analista de comunicação.

[Mais de Rafael H.](#)





Tutoriais relacionados

26 mar • [SSH](#)

[Como Configurar SSH Sem Senha no Linux](#)

SSH significa Secure Shell. É um protocolo de internet de código aberto que pode ser utilizado para acessar servidores e executar comandos...

[Por Arianne G.](#)

14 fev • [SSH](#)

[Como Compactar e Descompactar Arquivos via SSH](#)

Na maioria dos casos, os gerenciadores de arquivos web tem certas limitações quando se trata de descompactar arquivos. Ou talvez você esteja...

[Por Arianne G.](#)

25 ago • [SSH](#)

[Como funciona o SSH](#)

ConteúdoO Que é SSHComo funciona o SSHEntendendo diferentes técnicas de criptografiaEntendendo diferentes técnicas de criptografiaNegociando...

[Por Arianne G.](#)

Comentários

[Faça um comentário](#)

toyota hải dương
abril 16 2018

[RESPONDER](#)

I think everything posted made a great deal of sense. But, what about this? what if you were to create a killer title? I ain't suggesting your content is not solid., however suppose you added a title to maybe grab folk's attention? I mean Lista de Comandos Básicos SSH is a little vanilla. You might look at Yahoo's front page and note how they create article headlines to grab viewers to click. You might add a video or a picture or two to grab people interested about everything've got to say. In my opinion, it would make your blog a little bit more interesting.

josedaltro
abril 16 2018

Thank you for your feedback, we are checking and optimizing all of our blog content. We are prioritizing this one. You can check later the changes that we have made. ;)

Deixe uma resposta

Comentário*

Nome*

Email*

🔒 Ao inserir seu e-mail, você concorda em receber a newsletter da Hostinger. Você pode cancelar a inscrição a qualquer momento. Confira nossa

Enviar

Esse site utiliza o Akismet para reduzir spam. [Aprenda como seus dados de comentários são processados.](#)



Somos uma provedora de hospedagem de sites e nossa missão é promover o sucesso a todos que querem começar sua jornada online. Para isso, buscamos sempre melhorar a tecnologia dos nossos servidores, oferecer um suporte profissional e tornar a experiência dos nossos clientes com a hospedagem de sites a melhor de todas.



HOSPEDAGEM

- [Hospedagem de Sites](#)
- [Servidor VPS](#)
- [Servidores de Minecraft](#)
- [Hospedagem Cloud](#)
- [Hospedagem WordPress](#)
- [Hospedagem de Email Profissional](#)
- [Hospedagem CMS](#)
- [Hospedagem de Loja Virtual](#)
- [Hospedagem cPanel](#)
- [Hospedagem Grátis](#)
- [Loja Virtual](#)
- [Criador de Sites](#)
- [Comprar Hospedagem](#)

DOMÍNIOS

- [Registro de Domínio](#)
- [Transferir Domínio](#)
- [Domínio Grátis](#)
- [Domínio .XYZ](#)
- [Domínio Barato](#)
- [Comprar SSL](#)
- [WHOIS](#)
- [Certificado SSL Grátis](#)
- [Pesquisa de Domínios](#)

AJUDA

- [Tutoriais](#)
- [Base de Conhecimento](#)
- [Denunciar Abuso](#)

INFORMAÇÕES

- [Status do servidor](#)
- [Programa de Afiliados](#)
- [Métodos de Pagamento](#)
- [Termos de Migração](#)
- [Prêmio](#)
- [Avaliações](#)
- [Preços](#)

SOBRE NÓS

- [Sobre a Hostinger](#)
- [Nossa Tecnologia](#)
- [Carreira](#)
- [Fale Conosco](#)
- [Blog](#)

LEGAL

- [Política de Privacidade](#)
- [Termos de Serviço](#)

