CONTENTS

RELATED

Como instalar o PostgreSQL no Ubuntu 20.04: [Guia de início rápido]

View

Como Instalar o Servidor Web Apache no Ubuntu 18.04 [Quickstart]

View

// Tutorial //

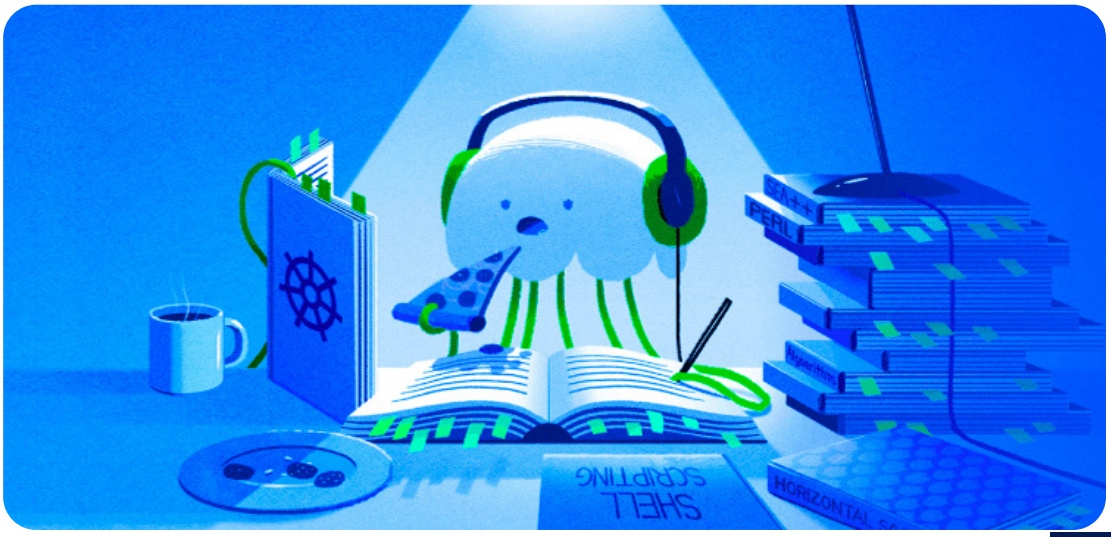
Configuração Inicial de servidor com Ubuntu 18.04

Published on May 16, 2018

Security Ubuntu Getting Started Initial Server Setup Ubuntu 18.04

By Justin Ellingwood and Erika Heidi

Português



Introdução

Quando você cria inicialmente um novo servidor Ubuntu 18.04, existem alguns passos de configuração que você deve tomar no início como parte da configuração básica. Isto aumentará a segurança e a usabilidade do seu servidor e dará uma sólida fundação para as ações subsequentes.

> **Nota:** O guia abaixo demonstra como completar manualmente os passos que recomendamos para novos servidores Ubuntu 18.04. Seguir esses processos manualmente pode ser útil para aprender algumas habilidades básicas de administração de sistemas e como um exercício para compreender plenamente as ações que estão sendo tomadas no seu servidor. Como uma alternativa, se você deseja começar a funcionar mais rapidamente, você pode executar o nosso script de configuração inicial de servidor que automatiza esses passos.

Passo Um - Efetuando login como Root

Para entrar no seu servidor, você precisa saber o **endereço IP público do servidor**. Você também vai precisar da senha ou, se você instalou uma chave SSH para autenticação, da chave privada para a conta do usuário **root**. Se você ainda não tiver conectado ao seu servidor, você pode querer seguir o nosso guia sobre Como se Conectar ao seu Droplet com SSH, que cobre esse processo em detalhes.

Se você ainda não estiver conectado ao seu servidor, vá em frente e acesse como usuário **root** utilizando o seguinte comando (substitua a parte realçada do comando com o endereço IP público do seu servidor):

```
$ ssh root@ip_do_seu_servidor
```

Copy

Aceite a mensagem de aviso sobre a autenticidade do host, se ela aparecer. Se você estiver utilizando autenticação por senha, forneça a sua senha de **root** para efetuar o login. Se você estiver utilizando uma chave SSH que esteja protegida por uma frase secreta, você pode ser solicitado a digitar a frase secreta na primeira vez em que usar a chave em cada sessão. Se esta é a primeira vez que você faz logon no servidor com uma senha, você também será solicitado a alterar a senha de **root**.

Sobre o Root

O usuário **root** é o usuário administrativo em um ambiente Linux que possui privilégios muito amplos. Devido aos privilégios elevados da conta **root**, você é desencorajado de utilizá-la regularmente. Isto é porque parte do poder inerente à conta **root** é a capacidade de realizar alterações muito destrutivas, mesmo por acidente.

O próximo passo é configurar uma conta de usuário alternativa com um escopo reduzido de poderes para o trabalho diário. Vamos ensiná-lo como obter aumento de privilégios durante os momentos em que você precisar deles.

Passo Dois - Criar um Novo Usuário

Uma vez conectado como **root**, estamos preparados para adicionar uma nova conta de usuário que utilizaremos para efetuar logon de agora em diante.

Este exemplo cria um novo usuário chamado **sammy**, mas você deve substituí-lo por um nome de usuário de sua escolha:

```
# adduser sammy
```

Copy

Você será solicitado a responder algumas perguntas, começando com a senha da conta.

Entre com uma senha forte e, opcionalmente, preencha quaisquer informações adicionais se desejar. Isso não é requerido e você pode apenas teclar `ENTER` em qualquer campo que você quiser pular.

Passo Três – Concedendo Privilégios Administrativos

Agora, temos uma nova conta de usuário com privilégios básicos de conta. Contudo, podemos às vezes precisar fazer tarefas administrativas.

Para evitar de ter que desconectar nosso usuário normal e efetuar login novamente com a conta de **root**, podemos configurar o que é conhecido como "super usuário" ou privilégios de **root** para nossa conta

normal. Isso irá permitir nosso usuário normal executar comandos com privilégios administrativos colocando a palavra `sudo` antes de cada comando.

Para adicionar esses privilégios para nosso novo usuário, precisamos adicionar o novo usuário ao grupo **sudo**. Por padrão, no Ubuntu 18.04, os usuários que pertencem ao grupo **sudo** estão autorizados a utilizar o comando `sudo`.

Como **root**, execute este comando para adicionar seu novo usuário ao grupo **sudo** (substitua a palavra em destaque pelo seu novo usuário):

```
# usermod -aG sudo sammy
```

Copy

Agora, quando estiver logado com seu usuário comum, você pode digitar `sudo` antes dos comandos para realizar ações com privilégios de super usuário.

Passo Quatro – Configuração de um Firewall Básico

Os servidores Ubuntu 18.04 podem utilizar o firewall UFW para certificar-se que somente conexões a certos serviços são permitidas. Podemos configurar um firewall básico muito facilmente utilizando esta aplicação.

> **Nota:** Se seus servidores estão rodando na DigitalOcean, você pode opcionalmente utilizar DigitalOcean Cloud Firewalls em vez do firewall UFW. Recomendamos a utilização de apenas um firewall de cada vez para evitar regras conflitantes que podem ser difíceis de depurar.

Diferentes aplicações podem registrar seus perfis com o UFW na instalação. Esses perfis permitem ao UFW gerenciar essas aplicações pelo nome. O OpenSSH, serviço que está nos permitindo conectar ao nosso servidor agora, tem um perfil registrado com o UFW.

Você pode ver isto ao digitar:

```
# ufw app list
```

Copy

```
Output
Available applications:
  OpenSSH
```

Precisamos ter certeza que o firewall permite conexões SSH de forma que possamos fazer login novamente da próxima vez. Podemos permitir essas conexões digitando:

```
# ufw allow OpenSSH
```

Copy

Depois, podemos ativar o firewall digitando:

```
# ufw enable
```

Copy

Digite "`y`" e pressione `ENTER` para prosseguir. Você pode ver que as conexões SSH ainda são permitidas ao digitar:

```
# ufw status
```

Copy

```
Output
Status: active

To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
```

Como **o firewall está bloqueando atualmente todas as conexões exceto por SSH**, se você instalar e configurar serviços adicionais, precisará ajustar as configurações de firewall para permitir o tráfego entrante. Você pode aprender algumas operações comuns do UFW nesse guia.

Passo Cinco – Ativando o Acesso Externo para o Seu Usuário Comum

Agora que temos um usuário comum para uso diário, precisamos ter certeza de que podemos acessar o SSH diretamente na conta.

> **Nota**: Até verificar se você pode efetuar login e usar `sudo` com seu novo usuário, recomendamos que você permaneça logado como **root**. Dessa forma, se você tiver problemas, você pode solucionar e fazer as alterações necessárias como **root**. Se você estiver utilizando um Droplet da DigitalOcean e experimentar problemas com sua conexão SSH de **root**, você pode efetuar o login no Droplet utilizando o Console da DigitalOcean.

O processo para configuração do acesso SSH para o seu novo usuário depende se a conta **root** do seu servidor usa senha ou chaves SSH para autenticação.

Se a Conta Root Usa Autenticação por Senha

Se você efetua login em sua conta **root** *utilizando uma senha*, então a autenticação por senha está ativada para o SSH. Você pode fazer SSH para sua nova conta de usuário através da abertura de uma nova sessão de terminal e da utilização do SSH com seu novo nome de usuário:

```
$ ssh sammy@ip_do_seu_servidor
```
Copy

Depois de entrar com a senha do seu usuário comum, você estará logado. Lembre-se, se você precisar executar um comando com privilégios administrativos, digite `sudo` antes do comando, dessa forma:

```
$ sudo comando_a_executar
```
Copy

Você será solicitado a digitar a senha para seu usuário comum ao utilizar o `sudo` pela primeira vez a cada sessão (e periodicamente depois).

Para aumentar a segurança do seu servidor, **recomendamos fortemente a configuração de chaves SSH em vez da utilização de autenticação por senha**. Siga nosso guia sobre configuração de chaves SSH no Ubuntu 18.04 para aprender como configurar autenticação baseada em chaves.

Se a Conta Root Usa Autenticação por Chave SSH

Se você estiver logado em sua conta **root** *utilizando chaves SSH*, então a autenticação por senha está desativada para o SSH. Você precisará adicionar uma cópia da sua chave pública local ao arquivo `~/.ssh/authorized_keys` do novo usuário para efetuar login com sucesso.

Uma vez que sua chave pública já está no arquivo `~/.ssh/authorized_keys` da conta **root** no servidor, podemos copiar esse arquivo e a estrutura de diretório para a nossa nova conta de usuário em nossa sessão atual.

A forma mais simples de copiar os arquivos com a propriedade e as permissões corretas é com o comando `rsync`. Isso irá copiar o diretório `.ssh` do usuário **root**, preservar as permissões, e modificar os proprietários do arquivo, tudo num único comando. Certifique-se de alterar as partes destacadas do comando abaixo para corresponder ao seu nome de usuário comum:

```
$ rsync --archive --chown=sammy:sammy ~/.ssh /home/sammy
```
Copy

Agora, abra uma nova sessão de terminal e utilize o SSH com seu novo usuário:

```
$ ssh sammy@ip_do_seu_servidor
```
Copy

Você deve estar logado na nova conta de usuário sem usar uma senha. Lembre-se, se você precisar

executar um comando com privilégios administrativos, digite `sudo` antes do comando, dessa forma:

```
$ sudo comando_a_executar
```

Copy

Você será solicitado a digitar a senha para seu usuário comum ao utilizar o `sudo` pela primeira vez a cada sessão (e periodicamente depois).

Para Onde ir a partir daqui?

Neste ponto, você tem uma base sólida para seu servidor. Você pode instalar qualquer software que você precisar em seu servidor agora.

About the authors

Justin Ellingwood Author

Developer and author at DigitalOcean.

Erika Heidi Author

Developer Advocate

Dev/Ops passionate about open source, PHP, and Linux.

Fernando Pimenta Translator

Developer and author at DigitalOcean.