Bare Metal Cloud is now live! DEPLOY TODAY

How To Enable SSH On Debian 9 Or 10

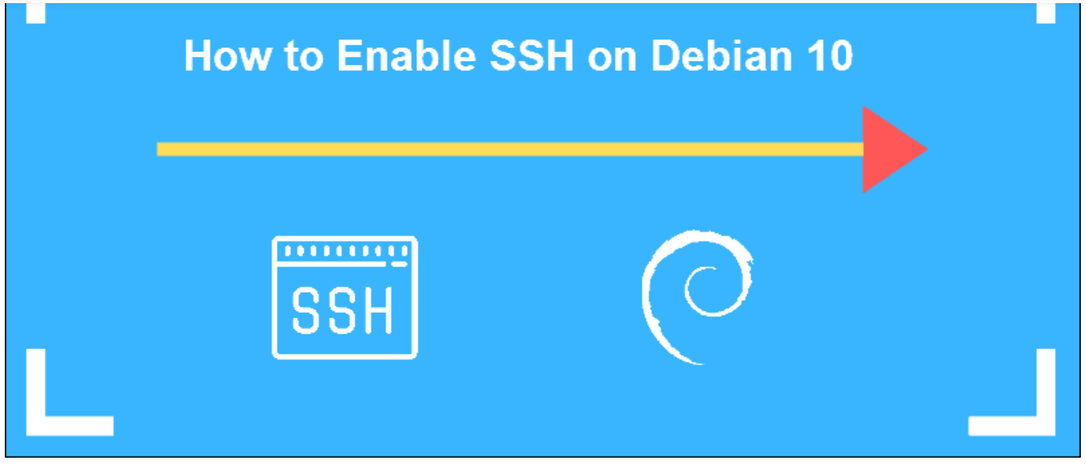
Posted July 30, 2019 SSH DEBIAN

Introduction

SSH stands for Secure Shell. SSH is used for connecting to a remote computer accessing files and perform administrative tasks.

In this tutorial, learn how to enable SSH on Debian 9 (Stretch) or Debian 10 (Buster).

Bare Metal Cloud is now live! DEPLOY TODAY



Prerequisites

- Debian system to act as an SSH server
- Debian system to act as an SSH client
- sudo privileges on Debian for each system
- Access to a command line (Ctrl-Alt-T)
- Apt package manager(included by default)

5 Steps to Enable SSH on Debian

When you're connecting remotely, a secure connection is important – without it, a hacker could intercept usernames, passwords, and configuration files that could compromise the security of your server. These five (5) steps will guide you throughout the process of establishing a secure connection.

Step 1: Update the Package Manager

Bare Metal Cloud is now live! DEPLOY TODAY

```
sudo apt-get update
```

The screen confirms that the packages have been updated:

```
phoenixnap@debian:~$ sudo apt-get update
Get:1 http://security.debian.org/debian-security buster/updates InRelease [39.1
kB]
Hit:2 http://deb.debian.org/debian buster InRelease
Get:3 http://deb.debian.org/debian buster-updates InRelease [46.8 kB]
Fetched 85.9 kB in 1s (91.9 kB/s)
Reading package lists... Done
```

Step 2: Install SSH Server

On the system that acts as a server, run the following command:

```
sudo apt install openssh-server
```

Enter your password when prompted, then press Y to continue the installation. In this case, the output states that the newest version is already installed.

```
phoenixnap@debian:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-server is already the newest version (1:7.9p1-10).
openssh-server set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

You can check the status of the SSH service with the following command:

```
sudo systemctl status ssh
```

Bare Metal Cloud is now live! DEPLOY TODAY

```
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enab
   Active: active (running) since Tue 2019-07-23 07:21:36 MST; 18h ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Process: 469 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 481 (sshd)
    Tasks: 1 (limit: 2348)
   Memory: 3.9M
   CGroup: /system.slice/ssh.service
           └─481 /usr/sbin/sshd -D
```

Step 3: Start and Stop the SSH Server

Since SSH controls connections, it can be handy to know how to start and stop the service.

To stop the SSH host server, enter the following:

```
sudo service ssh stop
```

If you check the status of the service at this point the system indicates that SSH is inactive. It also indicates the exact date and time it stopped.

```
phoenixnap@debian:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enab
   Active: inactive (dead) since Wed 2019-07-24 01:29:32 MST; 4s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
```

To start the SSH service, use the following command:

```
sudo service ssh start
```

Bare Metal Cloud is now live! DEPLOY TODAY

To turn off SSH indefinitely, enter:

```
sudo systemctl disable ssh
```

To re-enable the SSH service, simply replace disable with enable.

Step 4: Get Your Server IP Address

If you're configuring a server locally, you can display the IP address from a terminal window with the following command:

```
ip a
```

The IP address will be in a format like this:

```
192.168.0.1
```

If you're connecting to a server that's already configured, you'll want to get the IP address from the server's administrator. Or, you can log in using the server's hostname or domain name.

Step 5: Install SSH Client Service (Optional)

If you're connecting to a server that's already set up, or you've completed the previous steps on your server, open a terminal window on your client system and update the package list:

```
sudo apt-get update
```

Bare Metal Cloud is now live! DEPLOY TODAY

```
sudo apt-get install openssh-client
```

This example has the newest version already installed.

```
phoenixnap@debian:~$ sudo apt-get install openssh-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-client is already the newest version (1:7.9p1-10).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Connecting to a Server Using SSH

Enter the following command to connect to the server using a secure shell:

```
ssh UserName@IPAddressOrHostname
```

Replace the UserName with the username of an authorized user on the server. After the @ sign, use the IP address from Step 4, or you can use the domain name. You can also specify a hostname if the server is configured to use one.

When connecting for the first time, the system may prompt you for confirmation. Type '**yes**' and then '**enter**'.

The remote system will prompt you for a password. Use the password that goes with the username you've provided.

The command prompt will change to **username@hostname**, indicating that the commands you are running are being executed on the remote server.

If you're connecting from a Windows system, you'll need to install a third-party utility like PuTTY.

Bare Metal Cloud is now live! DEPLOY TODAY

Firewall and Security Settings

By default, Debian uses the UFW firewall which can interfere with secure shell traffic.

To allow SSH access, use the command:

```
sudo ufw allow ssh
```

SSH traffic goes through port 22. The output confirms that the rules have been updated.

```
phoenixnap@debian:~$ sudo ufw allow ssh
[sudo] password for phoenixnap:
Rules updated
Rules updated (v6)
```

To implement the necessary security measures, use your firewall application (or router configuration) to set up port forwarding. You'll need to refer to your documentation for specifics. However, the strategy is to forward traffic requests coming in on port 22 to the IP address of the machine behind the firewall.

You can also configure your firewall or router to accept SSH traffic on a different port, it's an added step, but then you can route that incoming traffic to port 22 on your server. This is a helpful solution when opening up your server to internet traffic. Why? Many intrusion attempts come in on port 22 trying which try to access SSH. Changing the port will restrict access to only those who know the correct port, thereby limiting unauthorized connections.

If you want to connect to a server on the internet, but you aren't sure what the IP address is, use this tool.

Bare Metal Cloud is now live! DEPLOY TODAY

By following the steps outlined in this article, you have successfully enabled an SSH connection on Debian.

You can now connect to a remote host securley and continue to manage your servers in a safe environment.

Check out our guide on "ssh_exchange_identification: Read: Connection Reset By Peer" error if you notice it while connecting to your remote server.

Next you should also read

Security, SysAdmin

5 Linux SSH Security Best Practices to Secure Your Systems

September 24, 2019

The article covers the 5 most common and efficient ways to secure an SSH connection. The listed solutions go…

READ MORE

Security, SysAdmin

19 Common SSH Commands in Linux With Examples

August 25, 2019

Secure Shell is an important protocol for anyone managing and controlling remote machines. This guide covers…

READ MORE

SysAdmin, Bare Metal Servers

How to upgrade Debian 9 Stretch to Linux Debian 10 Buster

March 29, 2019

Debian 10, codenamed Buster, is still a work in progress, but a pre-release version is available. Even though…

READ MORE

Security, Web Servers, Bare Metal Servers

How to Use SSH to Connect to a Remote Server in Linux or Windows

September 24, 2018

In this tutorial, Find out How To Use SSH to Connect to a Remote Server in Linux or Windows. Get started with…

READ MORE

Bare Metal Cloud is now live! DEPLOY TODAY

Author

Vladimir Kaplarevic

Vladimir is a resident Tech Writer at phoenixNAP. He has more than 7 years of experience in implementing e-commerce and online payment solutions with various global IT services providers. His articles aim to instill a passion for innovative technologies in others by providing practical advice and using an engaging writing style.

© 2020 Copyright phoenixNAP | Global IT Services. All Rights Reserved.