# How To Install/Enable SSH Server In Windows 10 and Windows Server 2016/2019?

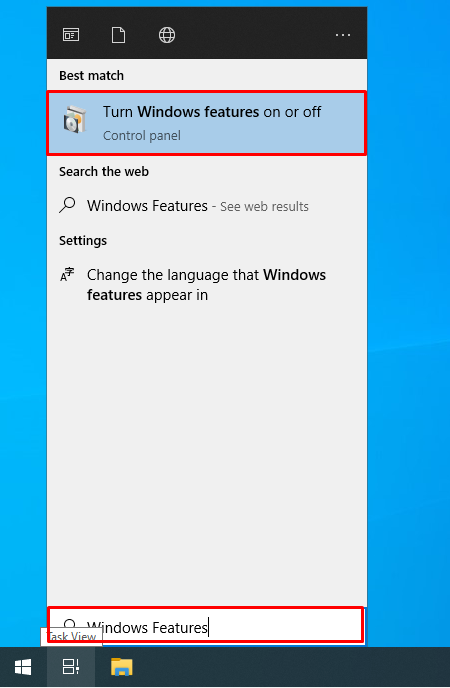
 by [İsmail Baydan](https://windowstect.com/author/ismail/)

**SSH**is a remote management protocol and tool which provides secure command-line access to the remote system. SSH is created for Linux and Unix systems but gained popularity on other systems like Network Routers, Network Switches, etc. Different 3d party software is created to run **SSH client** and **SSH server** on Windows operating systems but they should be installed externally from other vendors. Recent versions of the Windows operating system also support SSH client and SSH server as built-in. In this tutorial, we will examine how to install or enable the built-in SSH server for Windows 10 and Windows Server 2016/2019.

## Install/Enable SSH Server From Windows Features (GUI)

SSH server or OpenSSH server is available by Windows 10 and Windows 2019 which should be installed or enabled via the **Windows Features**. This can be done with 2 methods. In this part, we will install the SSH server by using the Windows Features GUI from the desktop. First, open the Windows Features with the following steps.

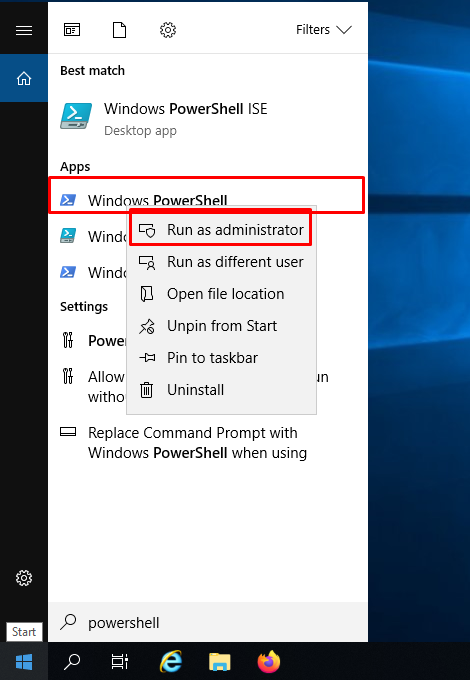
Type “**Windows Features**” to the Start Menu which will list the windows features shortcut.



Alternatively you can follow **Settings**-> **Apps**-> **Apps and Features** -> **Manage Optional Features** . Find the **OpenSSH Client** and **OpenSSH Server** in tick them. Then click the Apply which will install both the SSH server and SSH client.

## Install/Enable SSH Server via PowerShell Command Line Interface

The Windows Features can be also installed from the PowerShell command-line interface by using the **Add-WindowsCapability** commands. We will use this command in order to install the SSH Server and Client. As the Windows Feature installation requires Administrative privileges the PowerShell should be opened with the Administrator account or Administrator privileges. First list the PowerShell from the **Start Menu** where select the “**Run as administrator**” like below.



The second step checking whether the SSH Server and Client are installed with the **Get-WindowsCapability** command like below.

Get-WindowsCapability -Online | ? Name -like '\*SSH\*'

This command list all currently installed or enable Windows Features and filter them with the “ssh” term where the SSH-related features will be listed below. We can also see that the available OpenSSH.Client and OpenSSH.Server packages version is 0.0.2.0 .

Name : OpenSSH.Client~~~~0.0.2.0

State : NotPresent

Name : OpenSSH.Server~~~~0.0.2.0

State : NotPresent

The State show whether it is installed or not. The “**NotPresent**” means it is not installed. As we can see from the output that both of the SSH server and SSH client is not installed.

[**See also  This Task Requires That The User Account Specified Has Log On As Batch Job Rights**](https://windowstect.com/this-task-requires-that-the-user-account-specified-has-log-on-as-batch-job-rights/)

Now we will install the SSH server and client by using the **Add-WindowsCapability** command line below. The SSH Server feature is named **OpenSSH.Server** and SSH Client feature is named as **OpenSSH.Server**.

Add-WindowsCapability -Online -Name OpenSSH.Client

Add-WindowsCapability -Online -Name OpenSSH.Server

When the SSH Server and SSH Client installation is completed the following output will be listed.

# Both of these should return the following output:

Path :

Online : True

RestartNeeded : False

## Configure Firewall Ports For SSH Server

Windows enables the local Firewall by default which will only accept a limited number of ports for security reasons. The SSH server port is not allowed by default where we should enable it. This can be done via the Windows Firewall GUI but using the PowerShell is more practical as running the following PowerShell command will open the SSH server port for communication. The configuration contains the SSH Server executable file which is located under the C:\System32\OpenSSH with the name of sshd.exe .

New-NetFirewallRule -Name SSH -DisplayName 'OpenSSH SSH Server Port' -Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22 -Program "C:\System32\OpenSSH\sshd.exe"

The SSH service port firewall configuration will output following informations.

Name : SSH

DisplayName : OpenSSH SSH Server Port

Description :

DisplayGroup :

Group :

Enabled : True

Profile : Any

Platform : {}

Direction : Inbound

Action : Allow

EdgeTraversalPolicy : Block

LooseSourceMapping : False

LocalOnlyMapping : False

Owner :

PrimaryStatus : OK

Status : The rule was parsed successfully from the store. (65536)

EnforcementStatus : NotApplicable

PolicyStoreSource : PersistentStore

PolicyStoreSourceType : Local

## Check SSH Server Service Status

After the installation and firewall configuration is complete we will check if the SSH server service is running. We will use the Get-Service command and provides the SSH service name which is sshd.

Get-Service sshd

## Start SSH Server Service

The SSH server service can be start with the Start-Service command of PowerShell easily.

Start-Service sshd

## Start SSH Server Service Automatically After Boot

Some services are started automatically without any command after boot. We can set the SSH server service to start automatically after boot with the following **Set-Service** command and **-StartupType “Automatic”** attributes and parameters.

Set-Service -Name sshd -StartupType 'Automatic'

## Uninstall/Disable SSH Server From PowerShell

If you do not need the SSH Server or Client you can uninstall or disable them which is the very same way as the installation. You can use the Windows Features screen by unticking the feature and then click on the “Apply” which will remove the SSH server and SSH client. An alternative way is using the PowerShell again with Administrator privileges which are explained in the installation part. We will use the **Remove-WindowsCapability** command and provide the **-Name** attribute as OpenSSH.Client and OpenSSH.Server. But we can also uninstall only the OpenSSH.Server though.

# Uninstall the OpenSSH Client

Remove-WindowsCapability -Online -Name OpenSSH.Client

# Uninstall the OpenSSH Server

Remove-WindowsCapability -Online -Name OpenSSH.Server

## Connect SSH Server

By using a GUI client like Putty or the built-in Windows ssh client we can connect to the Windows SSH Server like below.

ssh ismail@192.168.10.10

## Uninstall OpenSSH Server in Windows via Windows Features (Optional Features)

The Windows built-in SSH server can be uninstalled by using the Windows Features screen. First open the Windows Features screen by typing “windows features” to the Start Menu. This is also described at the start of this tutorial. Then navigate to the “OpenSSH Server” line and uncheck the checkbox.

[**See also  How To List Files In Command Prompt In Windows?**](https://windowstect.com/how-to-list-files-in-command-prompt-in-windows/)

[Administration](https://windowstect.com/category/administration/), [PowerShell](https://windowstect.com/category/powershell/), [Windows 10](https://windowstect.com/category/windows-10/), [Windows 8](https://windowstect.com/category/windows-8/), [Windows Server](https://windowstect.com/category/windows-server/)[o](https://windowstect.com/tag/openssh/)penssh, [ssh](https://windowstect.com/tag/ssh/), [ssh client](https://windowstect.com/tag/ssh-client/), [ssh server](https://windowstect.com/tag/ssh-server/)

Categories

Tags

Post navigation

[What Is PowerShell.exe and Its Usage Examples?](https://windowstect.com/what-is-powershell-exe-and-its-usage-examples/)

[Windows 10 SSH (OpenSSH) Tutorial](https://windowstect.com/windows-10-ssh-openssh-tutorial/)