SSH Access

For cPanel & WHM version 58

(Home >> Security >> SSH Access)

Overview

This interface provides information about how to connect to another web server via the SSH (secure shell) network protocol.

The SSH (secure shell) network protocol allows you to connect to another web server over the Internet via a command line interface (CLI). You can use this network protocol to remotely manage your server, configure CGI scripts, and perform other tasks.

Many modern operating systems, such as Mac® OS X and Linux distributions, include SSH. If you use Microsoft Windows® to connect to your server, you **must** use an SSH client, such as PuTTY, to log in to your server.

Many Unix-based operating systems include standardized commands. For a list of standardized Unix-based (POSIX) commands, read the One-Serve website documentation.

> **Note:**
> Not all hosting providers allow shell access.

Connect to your server via SSH

The following sections describe how to connect to your server via various SSH clients.

PuTTY PuTTY and a private key Mac OS or Linux

To use PuTTY to connect to your server via SSH, perform the following steps:

1. Download and install the PuTTY client.
2. From the Windows *Start* menu, open the client.
3. In the *Session* interface, enter the hostname or IP address of the server in the *Host Name (or IP address)* text box.
4. Enter the Port number in the *Port* text box.

> **Note:**
> Make **certain** that you select the *SSH* protocol.

5. Click *Open*.
6. Enter `root` as the login name.
7. Enter the `root` password.

To in to a server via SSH with PuTTY and a public key, perform the following steps:

1. From the Windows *Start* menu, open the client.
2. Navigate to the *PuTTY Key Generator* interface.
3. Under the *Actions* heading, click *Generate*. PuTTY generates the key and displays the result under the *Key* menu.
4. Copy the public key and paste it in the `.ssh/authorized_keys` file.
5. Enter a passphrase in the *Key passphrase* and *Confirm passphrase* text boxes.
6. Click *Save private key* and save the key as a `.ppk` file.

> **Important:**
> You **must** save PuTTY keys as `.ppk` files.

7. In the *Session* interface, from the *Saved Sessions* menu, select your preferred

authorization session and click *Load*.
8. Navigate to the *Auth* interface under the *SSH* category.
9. Click *Browse*, select the private key file to upload, and click *Open.*
10. Navigate to cPanel's *Manage SSH Keys* interface (*Home >> Security >> SSH Access >> Manage SSH Keys)* and import the server's keys.

To log in to your Mac OS X or Linux server via SSH, perform the following steps:

1. Open a terminal session.
2. Run the following command:

```
ssh -p port user@IP
```

> **Note:**
> In the above command:
>
> - `port` represents the port number.
> - `user` represents your username.
> - `IP` represents your IP address.

Manage SSH keys

This section of cPanel's *SSH Access* interface allows you to create, import, manage, and remove SSH keys. The system will use these keys when you confirm that a specific computer has the right to access your website's information with SSH. You may perform the following actions from cPanel's *Manage SSH Keys* interface (*Home >> Security >> SSH Access >> Manage SSH Keys).*

Generate a New Key

Use this section of the interface to create new SSH key pairs, which include a public key and a private key.

To generate a new SSH key pair, perform the following steps:

1. Click *Generate a New Key*.
2. To use a custom key name, enter the key name in the *Key Name (This value defaults to id_rsa):* text box.

> **Note:**
> If you use a custom key name, you **must** manually specify the SSH key when you log in to the server.

3. Enter and confirm the new password in the appropriate text boxes.

> **Notes:**
> - The system grades the password that you enter on a scale of 100 points. `0` indicates a weak password, while `100` indicates a very secure password.
> - Some web hosts require a minimum password strength. A green password *Strength* meter indicates that the password is equal to or greater than the required password strength.
> - Click *Password Generator* to generate a strong password. For more information, read our Password & Security do cumentation.

4. Select the desired key type.
   - *DSA* keys provide quicker key generation and signing times.
   - *RSA* keys provide quicker verification times.
5. Select the desired key size.

> **Note:**
> Greater key sizes are more secure, but they result in larger file sizes and slower authentication times.

6. Click *Generate Key*. The interface will display the saved location of the key.

> **Important:**

For the new SSH key to function, you **must** authorize the SSH key. For more information, read the Manage your keys section.

Import Key

To import an existing SSH key, perform the following steps:

1. Click *Import Key*.
2. To use a custom key name, enter the key name in the *Choose a name for this key (defaults to id_dsa)* text box.

   Important:
   If you use a custom key name, you **must** manually specify the SSH key when you log in to the server.

3. Paste the public and private keys into the appropriate text boxes.
4. Click *Import*.

Manage your keys

The *Public Keys* and *Private Keys* tables display the following information about your existing keys:

Column	Description
Name	The key's name. Public and private keys use the same key name.
Authorization Status	Whether you authorized the key. **Important:** You **must** authorize new keys before you attempt to use them. **Note:** This column **only** displays in the *Public Keys* table.
Actions	You can perform the following actions: • *Delete Key* — Click to delete the key, and then click *Yes* to confirm that you wish to delete the key. • *View/Download Key* — Click to view or download the key. To download the key, save the contents of the *Public SSH Key* text box to your computer. • *Manage* — Click to manage authorization for the key. A new interface will appear. Click *Authorize* to authorize the key, or *Deauthorize* to revoke authorization for the key. **Notes:** • You can **only** perform this action for public keys. • After you deauthorize a key, that key's users **cannot** log in with the associated private key.