

Different types of SSH authentication keys

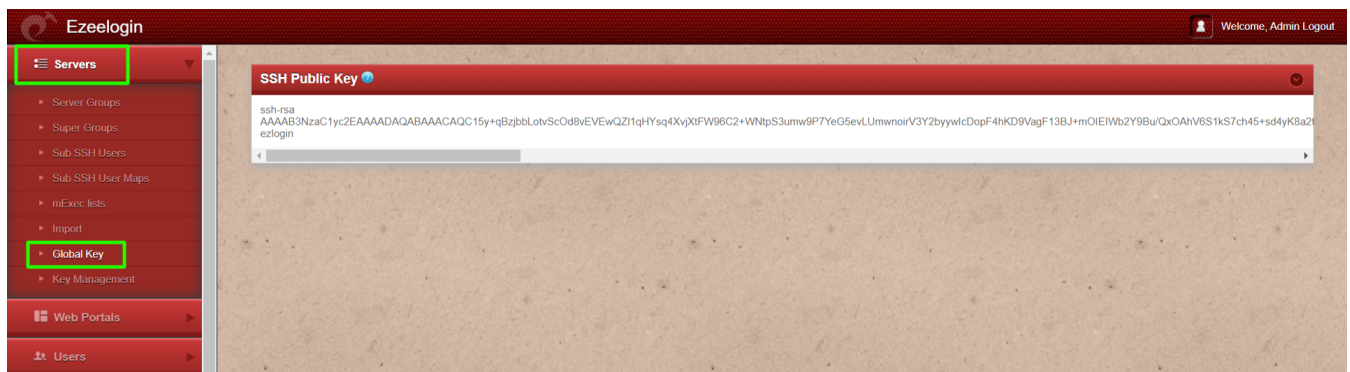
698 Jisna Joseph April 17, 2024 [Features & Functionalities](#) 158

What are the various types of SSH authentication keys used in Ezeelogin?

This article serves as a comprehensive guide to the various types of SSH authentication keys utilized in Ezeelogin and their respective purposes.

1. Ezeelogin Public Key (Global key)

Ezeelogin's public key or global key is the **ssh public key** used by the Ezeelogin gateway server to access the remote servers. The private key corresponding to the public key is encrypted and stored in Ezeelogin and cannot be retrieved. The private key for the global key is located at `/usr/local/etc/ezlogin/id_key` and the public key is located at `/usr/local/etc/ezlogin/id_key.pub` which is also visible in Ezeelogin software GUI under **Servers -> Global Key**. Global key which is created at the time of Ezeelogin installation only supports **RSA keys** and can only be [regenerated](#). The key that is regenerated will also be RSA key. You can also use [custom key pair](#) during installation only.



2. Ezeelogin Cluster Key

Cluster keys in Ezeelogin serve dual purposes serving both as the means for **communication between master-slave** servers through SSH and also as the keys utilized for **adding users** in Ezeelogin. These keys are organized such that the private key is located at `/usr/local/etc/ezlogin/id_clkey`, with its corresponding public key residing at `/usr/local/etc/ezlogin/id_clkey.pub`. Ezeelogin cluster keys only support the **RSA format**. Cluster keys can be [regenerated](#) which will be also in **RSA format**.

KEY USAGE IN EZEEOLOGIN

1.SERVER ADD KEYS

- ADD SERVER USING SSH KEY PAIR

You can provide the **SSH Private key** (corresponding public key should be added to users authorized_keys of server you are trying to add) which helps you to SSH to the **server without password**. Refer detailed article to add a server using ssh key pair in Ezeelogin.

[Add a server using ssh key pair in ezeelogin](#)

- **KEY MANAGEMENT**

You can add the **custom key** in **key management** under **Servers tab >> Key Management**. On adding the custom private keys in Key management, we can later **select the private key** while adding a server. Refer to the **3rd step in the below article**.

[Key Management](#)

2.USER SSH KEYS

- **UNMANAGED SUBSSH USER**

You can set up **key-based authentication** for unmanaged SubSSH users by specifying the **private key** and **passphrase** of the SubSSH user. Ensure that the corresponding public key is added to the **SubSSH user's authorized key file** on the **remote server**. Refer detailed article below.

[Unmanaged SubSSH User](#)

- **PASS THROUGH USER**

You can configure **key-based authentication** for pass-through users by providing the **private key** and **passphrase** of the pass through user. Make sure to add the corresponding public key to the **pass-through user's authorized key file** on the **remote server**. Refer to the detailed article below.

[Pass Through User](#)

- **Custom key for passwordless authentication for gateway user**

SSH gateway users can add their **SSH public Keys** for **passwordless authentication** to access the ezsh shell (Ezeelogin backend shell). To enter your public keys, press F2 key in the Ezeelogin shell. These keys can be of any type and will be added to the **authorized_keys** file of the **gateway user** at **/home/ezeeloginuser/.ssh/authorized_keys**. This allows users to access the shell **without being prompted for a password**. Refer to the below article for more details.

[How to add ssh public key for passwordless authentication in ssh](#)

3.SERVER IMPORT KEYS

- To import servers from a CSV file with a **custom key**, add **private keys** along with **passphrases** in the **Key Management** section under **Servers tab >> Key Management**. Refer to the **2nd part of the below article** .

[Import Server from CSV with Custom Key](#)

Online URL: <https://www.ezeelogin.com/kb/article/different-types-of-ssh-authentication-keys-698.html>